Explicit time integration with lumped mass matrix for enriched finite elements solution of time domain wave problems

M. Drolia[1], M.S. Mohamed[1], O. Laghrouche[1,*], M. Seaid[2], and A. El Kacimi[3]

[1]Institute for Infrastructure and Environment, Heriot-Watt University, Edinburgh EH14 4AS, UK
[2]Department of Engineering, Durham University, Durham DH1 3LE, UK
[3]Laboratory of Modelling and Combinatorial, FP Safi, Cadi Ayyad University, Morocco

Abstract

We present a partition of unity finite element method for wave propagation problems in the time domain using an explicit time integration scheme. Plane wave enrichment functions are introduced at the finite elements nodes which allows for a coarse mesh at low order polynomial shape functions even at high wavenumbers. The initial condition is formulated as a Galerkin approximation in the enriched function space. We also show the possibility of lumping the mass matrix which is approximated as a block diagonal system. The proposed method, with and without lumping, is validated using three test cases and compared to an implicit time integration approach. The stability of the proposed approach against different factors such as the choice of wavenumber for the enrichment functions, the spatial discretization, the distortions in mesh elements or the timestep size, is tested in the numerical studies. The method performance is measured for the solution accuracy and the CPU processing times. The results show significant advantages for the proposed lumping approach which outperforms other considered approaches in terms of stability. Furthermore, the resulting block diagonal system only requires a fraction of the CPU time needed to solve the full system associated with the non-lumped approaches.

Keywords. Wave Equation; Finite Elements; Partition of Unity; Time Domain; Field Enrichment; Lumped Mass

1 Introduction

In this paper we look into the solution of the wave equation using an enriched finite element method and an explicit time integration scheme with a lumped mass matrix. The problem is relevant to a vast range of applications such as acoustic scattering, mobile phones and MRI-scanners and arises frequently in designing novel devices [1, 2]. For example the design of waveguides and antennas are carried out routinely in the laboratories to aid in the design of transceiver circuits used in mobile phones [3,4]. The problem also arises in the design of concert halls to provide best acoustic experience. Solving such applications with deterministic approaches, requires discretization of the governing partial differential equations and a solution of the resulting system of algebraic equations. It is known that a deterministic approach provides a unique solution to a well-posed problem associated with given boundary and initial conditions. However, such an approach can be computationally demanding, in cases that require the solution of a very large number of unknown variables.
Different numerical approaches are available for solving wave problems. In the finite difference method the differential equation is approximated as a difference equation. This standard method was first developed in [5] for initial boundary value problems of Maxwell's equations and later for electromagnetic scattering problems [6, 7]. The Boundary Element Method is another approach that is based on the derivation of variational boundary integral equations. Applications for Maxwell's equations on Lipschitz surfaces can be found in [8]. The Galerkin boundary element approach was developed in [9,10] for

simulation of electromagnetic scattering in the frequency domain. The finite element method (FEM) is also an alternative that simplifies complicated geometries and non-homogeneous media. The solution space is constructed with special basis functions. The coefficients of these basis functions are evaluated by solving a system of linear equations forged from the governing equation. In [11, 12], the FEM is described for wave propagation, scattering and radiation in periodic structures in the context of wideband antennas and transient electromagnetic phenomena.

One main issue with deterministic numerical methods is the large number of unknowns required for solving wave problems within a given engineering accuracy. The reason for this could be derived from the Nyquist sampling theorem, that sets a minimum number of sampling points (and hence the sampling frequency) required to capture a wave. This often leads to models which have millions of unknown variables to be determined [13, 14]. A major technique to overcome this drawback is to use special functions to enrich the deterministic approach approximation space. The partition of unity method which is based on enrichment with exponential functions is an example. It was first proposed for solving time-harmonic wave problems governed by the Helmholtz equation [15]. The idea is to enrich the approximation space with oscillatory functions. The partition of unity finite element method (PUFEM) was developed for solving the Helmholtz problems at high frequencies [16,17] and to deal with problems where the wave speed is piecewise constant [18]. The method outperforms the FEM with the standard polynomials basis functions [17,19]. The partition of unity enrichment technique led to a vast amount of literature comprising different approaches such as the generalized finite element method [20,21], the ultra-weak variational formulation [22,23], as well as the discontinuous enrichment method [24–28]. The PUFEM is also adapted for solving forward problems for heterogeneous materials [29], elastic waves [30,31] and more recently for inverse problem [32]. In [33–35] the authors adopt the PUFEM with implicit integration in time for solving various transient diffusion problems. The reader is also referred to [36] for a recent survey on different approaches that use some form of enrichment.

The wave propagation can be recovered by solving the partial differential equations using either the frequency- or the time-domain analysis. The above work on enrichment methods are developed for frequency domain problems. However wave equations in the time domain can deal with a much wider range of applications than its frequency-domain counterpart. For example pulse sources cannot be modelled by the frequency domain which require harmonic sources. The frequency domain approach also fails in cases which require a large range of frequencies such as in radar applications. To deal with time-domain wave problems, an enriched model for wave propagation was developed for two-dimensional transient wave problems in [37]. A PUFEM approach for time-domain electromagnetic waves was also developed in [38] using an implicit time stepping method. In [39], the authors present the usage of numerical manifold method to construct finite element patches that cover the numerical domain and produce diagonalised lumped mass matrices that satisfy mass conservation of the elements for elastodynamic problems. In [40], an explicit scheme for discontinuous deformation analysis is presented for elasticity problems. Similarly, in [41] a multi-patch approach is used with discontinuous Galerkin isogeometric analysis to form block diagonalised lumped mass matrices for wave propagation. The presented numerical analysis for transient hyperbolic problems shows that the step-size in time scales as $O(h/p)$ for stability where h is the mesh size and p is the order of approximation. An explicit time stepping scheme with lumped mass matrix can be highly beneficial for enrichment methods when solving wave problems. However, this was still not achieved. In general high order numerical schemes for time domain integration are limited by the fact that they have more restrictive Courant–Friedrichs–Lewy (CFL) conditions and suffer from challenging memory requirements. They are derived from the Taylor series expansion of the derivatives and assuming coefficients from multiple previous time steps [42]. In these methods, the mass matrix could pose a problem for computation as it would be required to be inverted in every time step. One way to reduce computational cost could be diagonalizing this matrix using the so called lumped mass approach. There are essentially a handful of ways to achieve a lumped mass matrix [43]. For instance one can use the nodal quadrature technique. In essence it requires usage of numerical integration points located at the nodes of each element. In the p-FEM, the banded mass matrix could be reduced to a diagonal matrix by the usage of Gauss-

Lobatto scheme for both interpolation and numerical integration. This is effective especially for third or higher order polynomials used in the finite element approximation. The only two criteria are that the order of accuracy must be maintained, and that the same points must be used for interpolation as for numerical integration i.e. including the end points [44]. However, due to the contribution of the enrichment functions it is not possible to diagonalise the mass matrix using similar quadrature techniques.
In [45], a variational approach is develolped to approximate the consistent mass matrix for partition of unity methods, which is in line with the classical row-summation technique for linear FEM. In our work we present a similar approach using plane wave enrichments, and investigate the benefits of mass lumping in terms of improved conditioning, and also in terms of performance over both structured and unstructured meshes. The contribution in this work is also derived from projecting arbitrary non-zero initial conditions over local approximation spaces for the enriched PUFEM. The formulation of block diagonal lumped mass matrix approach significantly simplifies the solution of the linear systems arising from the spatio-temporal discretizations of the time-domain wave problems. The mass matrix lumping also allows to improve the conditioning of such systems which are known to be ill conditioned. We focus on the difficulties related to capturing the oscillatory nature of solutions to wave phenomenon, such as the Maxwell equations with a source term (i.e. a non-zero right-hand side). The results obtained herein can be applied to other fields such as acoustics with the same basic principles, superimposed in a linear manner. Although developed for the PUFEM method, this lumping approach can be also used in other enrichment techniques without significant changes. The rest of this paper is organised as follows. First, the problem and the considered finite element method are defined. Details about the time integration schemes and the lumping of the mass matrix are provided in section 2. Next, several numerical experiments are presented in section 3. Finally, section 4 contains some concluding remarks and recommendations for future work.

2 Wave problem and FEM approximations

First, we present a linear wave equation in two dimensions which would be the basis of all analysis presented in this work. We define an initial boundary value problem on $\Omega \subset \mathbb{R}^2$ being an open bounded domain with Lipschitz continuous boundary Γ evolving in $[0, T]$ which is the time interval for the wave propagation. The problem is defined as

$$\frac{1}{c^2}\frac{\partial^2 E}{\partial t^2} - \nabla^2 E = f(t,\mathbf{x}), \qquad (t,\mathbf{x}) \in [0,T[\times\Omega, \tag{1a}$$

$$\frac{\partial E}{\partial \hat{\mathbf{v}}} + hE = g(t,\mathbf{x}), \qquad (t,\mathbf{x}) \in [0,T[\times\Gamma, \tag{1b}$$

$$E(0,\mathbf{x}) = E^0(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \tag{1c}$$

$$\frac{\partial E}{\partial t}(0,\mathbf{x}) = V^0(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \tag{1d}$$

where $\mathbf{x} = (x, y)^\top$ are the Cartesian coordinates, t is the time variable, $\hat{\mathbf{v}}$ the outward unit normal on Γ and E the magnitude of the transverse electric field in the direction perpendicular to the plane of numerical domain while c and h are constants. The functions $f(t,\mathbf{x})$ and $g(t,\mathbf{x})$ in (1) are the prescribed source and boundary functions, respectively. The functions $E^0(\mathbf{x})$ and $V^0(\mathbf{x})$ denote the given initial conditions. This model can be used to represent various linear electromagnetic and acoustic wave propagation problems. For instance, applied to the scalar field in a transverse mode of electromagnetic wave propagation, it can represent an accurate and efficient solution for a short pulse propagating over long distances.
We reduce the second order differential equation of (1a) to a system of first order derivatives as follows

$$\frac{\partial E}{\partial t} = V, \tag{2a}$$

$$\frac{1}{c^2}\frac{\partial V}{\partial t} = f(t,\mathbf{x}) + \nabla^2 E. \tag{2b}$$

For discretization in space, we approximate the solution space as a linear sum of traditional, class $C^0(\Omega)$, hat functions denoted by $\mathcal{N}(\mathbf{x}) = (\mathcal{N}_1, \cdots, N_{N_d})^\top$ characterized by the property $\mathcal{N}_i(\mathbf{x}_j) = \delta_{ij}$ with δ_{ij} denoting the Kronecker symbol. These hat functions are, as usual, defined over the finite elements $\mathcal{T}_i$ used to discretize the spatial domain Ω. Hence, the field E and its first derivative V are defined in the partition Ω_h as

$$E_h(t,\mathbf{x}) = \sum_{j=1}^{N_d} E_j(t)\mathcal{N}_j(\mathbf{x}), \quad V_h(t,\mathbf{x}) = \sum_{j=1}^{N_d} V_j(t)\mathcal{N}_j(\mathbf{x}), \tag{3}$$

where N_d is the total number of nodal points in the partition and h is the discretization parameter of the mesh. Thus, $E_j(t)$ and $V_j(t)$ are the nodal values of the global functions $E_h(t,\mathbf{x})$ and $V_h(t,\mathbf{x})$, respectively. Note that only the nodal values of the field (or its derivative) are functions of time t at a given time. This is just to signify that the coefficients of the hat functions depend on time, and that the hat functions themselves have no dependency on time whatsoever.
Hence a weak form is obtained by multiplying both sides of (2) with a weighting function $\phi(\mathbf{x})$ and integrating on both sides over Ω. Using the divergence theorem and using the boundary condition (1b) we obtain the following weak formulation of the problem (1)

$$\int_\Omega \frac{\partial E}{\partial t}\phi \, d\Omega = \int_\Omega V\phi \, d\Omega, \tag{4a}$$

$$\frac{1}{c^2}\int_\Omega \frac{\partial V}{\partial t}\phi \, d\Omega = -\int_\Omega \nabla E\cdot\nabla\phi \, d\Omega - \oint_\Gamma (hE)\phi \, d\Gamma + \oint_\Gamma g(t,\mathbf{x})\phi \, d\Gamma + \int_\Omega f(t,\mathbf{x})\phi \, d\Omega. \tag{4b}$$

Following a partition of unity approximation, each mesh node is enriched with plane waves, thus allowing for a much coarser mesh, and overall a significantly lower total number of degrees of freedom. Thus, the discretization equations in (3) are re-written as

$$E_h(t,\mathbf{x}) = \sum_{j=1}^{N_d}\sum_{q=1}^{Q} \hat{E}_j^q(t)e^{\mathrm{i}kz_q}\mathcal{N}_j(\mathbf{x}), \tag{5a}$$

$$V_h(t,\mathbf{x}) = \sum_{j=1}^{N_d}\sum_{q=1}^{Q} \hat{V}_j^q(t)e^{\mathrm{i}kz_q}\mathcal{N}_j(\mathbf{x}), \tag{5b}$$

where $z_q = x\cos\alpha_q + y\sin\alpha_q$ for $q = 1,\ldots,Q-1$ and $z_Q = 0$. Notice that the solution E is split into two components where $\hat{E}_j^Q$ represents a scaling factor for the nodal shape functions while $\hat{E}_j^q|_{q\neq Q}$ represents the amplitude of different plane waves used as nodal enrichments. Similar argument follows for its first derivative V. Thus, each node has exactly Q degrees of freedom, of which $Q-1$ are the amplitudes of plane wave enrichments, and the domain Ω_h has a total of N_dQ degrees of freedom. For the numerical integration of these plane wave enrichment functions over the spatial domain, we use about 10 integration points per wavelength in each direction. In [46], an explicit closed-form solution for two-dimensional wave-based integrals are developed, for the PUFEM using pressure and shear plane waves for local enrichment. Similar approach could be developed for the work presented in this paper, that could significantly reduce the computational cost associated with the assembly process. The constant k is problem dependent, albeit it is possible to run a parametric sweep in a selected (discrete) bandwidth to obtain a modal response, however this is beyond the scope of the current

work. The angles α_q for the plane wave enrichment functions are selected by uniformly dividing 2π based on $Q-1$. For example, $Q-1=4$ leads to $\alpha_q = 0$, $\pi/2$, π and $3\pi/2$ as the four distinct angles for the plane wave enrichments.
For the time integration we set two unknown vectors $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$ that store the time dependent coefficients, i.e.

$$\mathbf{y}_1(t)|_j = \{\hat{E}_j^1(t), \hat{E}_j^2(t), \ldots, \hat{E}_j^Q(t)\}^\top, \quad \mathbf{y}_2(t)|_j = \{\hat{V}_j^1(t), \hat{V}_j^2(t), \ldots, \hat{V}_j^Q(t)\}^\top, \tag{6}$$

where the subscript j denotes reference to a node in the partition Ω_h. Note that both $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$ are each of total size QN_d. Using, the definitions given in (5) and substituting them in (4) we obtain the following algebraic system of ordinary differential equations

$$\mathbf{M}\frac{d\mathbf{y}_1}{dt} = \mathbf{M}\mathbf{y}_2, \tag{7a}$$

$$\frac{1}{c^2}\mathbf{M}\frac{d\mathbf{y}_2}{dt} = (-\mathbf{K} - \mathbf{M}_\Gamma)\,\mathbf{y}_1 + \mathbf{f}_\Gamma + \mathbf{f}_\Omega. \tag{7b}$$

Here, the vectors and matrices in (7) are defined as follows:

$$\mathbf{M}_{(i,n)(j,m)} = \int_\Omega \mathcal{N}_i e^{\mathrm{i}kz_n}\mathcal{N}_j e^{\mathrm{i}kz_m} d\Omega, \tag{8a}$$

$$\mathbf{K}_{(i,n)(j,m)} = \int_\Omega \left(\nabla\mathcal{N}_i e^{\mathrm{i}kz_n}\right)\cdot\left(\nabla\mathcal{N}_j e^{\mathrm{i}kz_m}\right) d\Omega, \tag{8b}$$

$$\mathbf{M}_{\Gamma(i,n)(j,m)} = h\oint_\Gamma \mathcal{N}_i e^{\mathrm{i}kz_n}\mathcal{N}_j e^{\mathrm{i}kz_m} d\Gamma, \tag{8c}$$

$$\mathbf{f}_{\Gamma(i,n)} = \oint_\Gamma g(t,\mathbf{x})\mathcal{N}_i e^{\mathrm{i}kz_n} d\Gamma, \tag{8d}$$

$$\mathbf{f}_{\Omega(i,n)} = \int_\Omega f(t,\mathbf{x})\mathcal{N}_i e^{\mathrm{i}kz_n} d\Omega, \tag{8e}$$

where $(i,n) \in \{1,2,\ldots,N_d\}\times\{1,2,\ldots,Q\}$ denote the row incides, and similarly (j,m) denote the column indices respectively. The matrix equations in (7) can be further simplified by multiplying both sides with the inverse of the mass matrix $\mathbf{M}$

$$\mathbf{M^{-1}M}\frac{d\mathbf{y}_1}{dt} = \mathbf{M^{-1}M}\mathbf{y}_2,$$
$$\frac{1}{c^2}\mathbf{M^{-1}M}\frac{d\mathbf{y}_2}{dt} = \mathbf{M^{-1}}\,(-\mathbf{K} - \mathbf{M}_\Gamma)\,\mathbf{y}_1 + \mathbf{M^{-1}}\,(\mathbf{f}_\Gamma + \mathbf{f}_\Omega),$$

which can be written in a compact form as

$$\frac{d\mathbf{y}_1}{dt} = \mathbf{y}_2, \tag{9a}$$

$$\frac{d\mathbf{y}_2}{dt} = \mathbf{A}\mathbf{y}_1 + \tilde{\mathbf{r}}, \tag{9b}$$

where $\mathbf{A} = c^2\mathbf{M^{-1}}\,(-\mathbf{K} - \mathbf{M}_\Gamma)$ and $\tilde{\mathbf{r}} = c^2\mathbf{M^{-1}}\,(\mathbf{f}_\Omega + \mathbf{f}_\Gamma)$, respectively. Equation (9) can be further simplified as the standard notation for a differential equation

$$\frac{d\mathbf{y}}{dt} = \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ \mathbf{A} & \mathbf{0}\end{bmatrix}\mathbf{y} + \begin{bmatrix}\mathbf{0}\\ \tilde{\mathbf{r}}\end{bmatrix}$$

or simply,

$$\frac{d\mathbf{y}}{dt} = \mathbf{B}\mathbf{y} + \mathbf{r}. \tag{10}$$

It is worth noting that the matrix $\mathbf{B}$ is independent of the time variable, whereas the vector $\mathbf{r}$ depends on time. Thus, with a difference scheme adopted for time discretization, one needs only to update the right hand side term $\mathbf{r}$ in (10) to solve in $\mathbf{y}$ for subsequent time steps.

In the current study, we present the analysis with the aid of the explicit Euler scheme to integrate in time. Thus, our difference equation is given as $\mathbf{y}^{n+1} = \mathbf{y}^n + \Delta t\,(\mathbf{B}\mathbf{y}^n + \mathbf{r}^n)$ where superscript n is the index of the discretized time and refers to the time instance $(n)\Delta t$ while Δt is the time step size. With this iterative scheme, one does run into the problem of obtaining the coefficient vector $\mathbf{y}^0$ pertaining to the given initial condition. The traditional way around this problem is to either (a) use nodal quadrature method and assign exact values at the nodes at time $t = 0$ as the coefficients or (b) obtain the best approximation of the initial condition over the discritization Ω_h. To the author's best knowledge, these best approximations are generally obtained for lagrangian basis functions. In this paper, we solve a Galerkin equation to obtain the best approximation for the initial condition over Ω_h using the enriched solution space as defined in equation (5). The corresponding weak forms are given as

$$\int_\Omega E_h(0,\mathbf{x})\phi \; d\Omega = \int_\Omega E^0(\mathbf{x})\phi \; d\Omega, \tag{11a}$$

$$\int_\Omega V_h(0,\mathbf{x})\phi \; d\Omega = \int_\Omega V^0(\mathbf{x})\phi \; d\Omega. \tag{11b}$$

This could be written compactly as $\mathbf{M}\mathbf{y}^0 = \mathbf{f}^0_\Omega$, where $\mathbf{M}$ is the same as in equation (7) and $\mathbf{f}^0_\Omega$ is effectively the right hand side of (11). Solving this linear system of equations gives us the desired vector of coefficients $\mathbf{y}^0$ to initialize the time-stepping scheme.

It should be noted that the integration in time of the expression (10) is directly achievable for a wide range of higher order explicit methods such as Runge-Kutta methods. However, in this paper only the first order Euler scheme is implemented where using higher orders leads to a larger memory requirement as the number of the stored time steps will increase. Considering that we run reference solutions with the conventional finite element method, such requirement can cause an extra burden on the computational resources. At the same time the comparisons between the PUFEM and the FEM remains to be valid as long as the same time integration scheme is implemented with both methods no matter whether the scheme is high or low order. Hence, we only use a first order time integration scheme.

In contrast to the assembly process associated with the proposed explicit scheme, the implicit method used for the PUFEM in [38] differs in its assembly mostly in the way the integral over Ω is evaluated. For the implicit method, the $\mathbf{M}$ and the $\mathbf{K}$ terms of (8) are evaluated and stored as a single term, as $\mathbf{M}^{\text{Imp}} = \mathbf{M} + \mathbf{K}$. Similarly, the the right-hand side $\mathbf{f}_\Omega$ for the implicit formulation is given as

$$\mathbf{f}^{\text{Imp}}_\Omega = \int_\Omega \Big(2E^{n-1} - E^{n-2} + \Delta t^2 f(t_n,\mathbf{x})\Big)\mathcal{N}^\top \; d\Omega, \tag{12}$$

where E^{n-1} and E^{n-2} are the solutions at two consecutively previous steps in time. The boundary integrals in both formulations are very similar. These differences are the direct cause of dissimilarities observed in the overall computational cost associated with the assembly process in the two methods, as discussed in section 3.1.3. For more details on the implementation of implicit time stepping methods for the wave equation we refer to [38] amongst others.

One of the attractive features of using an explicit integration in time is the advantage of exploiting the diagonalization of the mass matrix. As it can be seen from the system equations in (9), one requires only to invert the mass matrix. The process of inversion can be significantly simplified by diagonalizing the mass matrix. In the FEM, one can achieve a purely diagonal mass matrix, by simply summing up all the columns of the mass matrix along each row into the respective diagonal. The off-diagonal terms are then set to zero. The approximate mass matrix for a linear FEM following

this classic row-summation thus is given as

$$\bar{\mathbf{M}}_{(i)(j)} = \begin{cases} 0, & i \neq j, \\ \sum_{l=1}^{N_d} \int_\Omega \mathcal{N}_i \mathcal{N}_l d\Omega, & i = j. \end{cases} \tag{13}$$

Thus, inverting this approximate mass matrix then simply requires inverting the diagonal terms, which is trivial. Since one only needs to store the diagonal terms, this also reduces the memory requirements for the mass matrix storage. Such a lumping procedure is rigorously developed for the partition of unity method, for non-negative weight functions and high-order local approximation space in [45], where the local approximation spaces are based on a mix of polynomials, singular and discontinious functions. In our work, we employ this classic row-summation for plane waves used as enrichment. Thus, in the case of our proposed PUFEM each node now contributes towards a $Q \times Q$ block in the mass matrix. Therefore we propose to its diagonalization in the same manner as in the standard FEM, which leads to a block-diagonal mass matrix. This is achieved by simply summing up all the off-diagonal blocks along each row into the respective diagonal block. The approximate mass matrix is given as

$$\bar{\mathbf{M}}_{(i,n)(j,m)} = \begin{cases} 0, & i \neq j, \\ \sum_{l=1}^{N_d} \int_\Omega \mathcal{N}_i e^{\mathrm{i}kz_n} \mathcal{N}_l e^{\mathrm{i}kz_m} d\Omega, & i = j. \end{cases} \tag{14}$$

This lumped mass matrix is much easier to store and invert as compared to the non-diagonalized consistent mass matrix where again the inversion of the mass matrix is now reduced to the inversion of smaller non-overlapping blocks of size $Q \times Q$. The total number of such blocks depends on the total number of nodes in the spatial mesh.

3 Numerical results and examples

In this section, we analyse the efficiency of the proposed methods, that is the PUFEM with explicit time integration, henceforth mentioned as PUFEM-E and its lumped mass matrix counterpart as PUFEM-BD. The results are also compared to the PUFEM with implicit time integration PUFEM-I. The relative error is evaluated in L^1-norm as

$$\text{Error} = \frac{\|E_{num} - E\|_{\mathrm{L}^1(\Omega)}}{\|E\|_{\mathrm{L}^1(\Omega)}} \times 100\%, \tag{15}$$

where E_{num} is the numerical solution obtained with each method while E is the exact solution.

3.1 Example of radial wave

First, we consider a circular wave with the source located at the system origin as studied in [38]. The analytical solution is given by

$$E(t, x, y) = e^{\mathrm{i}(kr - \omega t)},$$

where $r = \sqrt{x^2 + y^2}$. A unit square computational domain is assumed $\Omega = [0.1, 1.1] \times [0.1, 1.1]$. The analytical solution is imposed on the domain boundary using (1b) and also used to evaluate the initial conditions (1c) and (1d). Three different wavenumbers are considered $k = 20\pi$, $k = 40\pi$ and $k = 80\pi$ with the angular frequency fixed at $\omega = 1$.

3.1.1 Timestep convergence

Our first concern in this example is to evaluate the convergence of the PUFEM using explicit time integration where the mass matrix is lumped or full. The domain is discretized into a uniform mesh of 4-noded square elements. Hence, the number of elements is 4 in each direction and the total number

Table 1: Convergence with refined timestep size in the three considered time integration schemes for the example of radial wave at final time $t = 1$ and uniform mesh of 4×4 elements. The relative percentage errors are shown in columns corresponding to PUFEM-I, PUFEM-E and PUFEM-BD methods respectively.

	$k = 20\pi$, $Q = 27$			$k = 40\pi$, $Q = 57$		
Δt	PUFEM-I	PUFEM-E	PUFEM-BD	PUFEM-I	PUFEM-E	PUFEM-BD
0.01	1.32	0.80	2.99	1.01	0.47	2.01
0.005	1.12	0.84	2.91	1.44	0.38	1.90
0.001	0.98	0.92	2.86	29.87	0.44	1.84

$k = 80\pi$, $Q = 97$		
PUFEM-I	PUFEM-E	PUFEM-BD
0.79	0.57	4.08
0.68	0.46	3.96
4.72	0.46	3.88

of elements is 16 while the total number of nodes is 25. For $k = 20\pi$ we enrich the PUFEM with $Q = 27$, whereas for $k = 40\pi$ and $k = 80\pi$, the chosen number of enrichment functions is $Q = 57$ and 97, respectively. Table 1 compares the errors of PUFEM-I, PUFEM-E and PUFEM-BD methods, for three different time steps Δt. The errors are displayed at time $t = 1$ for the three methods.
It is clear that both PUFEM-E and PUFEM-BD methods converge with respect to the timestep refinement where smaller errors are obtained with smaller values of Δt. For $k = 20\pi$. both PUFEM-E and PUFEM-I methods give practically similar accuracy, whereas PUFEM-BD method leads to larger errors due to the approximation of the block diagonalization of the mass matrix. However, as a higher wavenumber is considered with a higher Q, PUFEM-I method shows significant increase in the error for the smallest considered timestep $\Delta t = 0.001$. The results suggest that PUFEM-I method is less stable than PUFEM-E and PUFEM-BD methods for the accumulation of the round-off errors while for the smallest Δt the solution requires 1000 steps to achieve the final time $t = 1$. PUFEM-I method seems also more sensitive to the ill-conditioning issue often observed at higher number of enrichment functions, see for instance [19, 47].
An explicit scheme is conditionally stable and usually requires the CFL number to be less than 1. On the other hand an implicit scheme is unconditionally stable and thus in general can tolerate a larger Δt. In a typical finite difference scheme, the CFL number is of the order $\sim \frac{u_m \Delta t}{h}$ where u_m is the maximum value of the wave speed present in the analytical solution and h and Δt are the space and time discretization constants, respectively. Heuristically speaking, the CFL number provides a measure of how fast the numerical value is calculated as opposed to the rate of change of the analytical solution on a given space-time grid. Hence, for the stability of the explicit approach a smaller Δt is required for a smaller space discretization. This can be a major issue for wave problems where highly refined meshes are necessary at high wavenumbers. However, thanks to the enrichment we show in this example that the same coarse mesh grid can be retained also when higher wavenumbers are considered. The coarse meshes are associated with relatively large values of the spatial step h. The CFL condition in this case becomes a trivial requirement unless Δt is significantly large. Hence, the sensitivity of the proposed explicit methods against Δt is quite similar to that of the reference implicit method PUFEM-I. This, allows utilizing the benefits of the explicit scheme (such as mass lumping and distributed computation) without compromising on the time step Δt for numerical stability.

3.1.2 Number of enrichment convergence

Our next aim in this example is to compare the convergence of the three methods when increasing the number of plane wave enrichments, which is often mentioned as the q-convergence, see for instance [20].

Table 2: The q-convergence for the three time integration schemes considered for the example of radial wave. Note that here a '–' replaces the results of a diverged case.

		Error (%)			Condition number		
k	Q	PUFEM-I	PUFEM-E	PUFEM-BD	PUFEM-I	PUFEM-E	PUFEM-BD
20π	21	1.84	1.81	7.64	6.13E+04	6.36E+04	1.34E+02
	31	2.17	0.89	1.61	5.65E+07	6.81E+07	9.40E+06
	41	–	–	1.20	6.43E+08	7.10E+08	2.94E+07
40π	37	3.79	3.78	10.92	3.56E+04	3.61E+04	5.93E+01
	47	0.8	0.76	4.67	9.8E+05	3.67E+06	1.72E+04
	57	–	0.43	1.84	2.09E+07	1.98E+08	1.43E+07
	67	–	–	1.00	3.47E+07	3.37E+07	1.66E+08
80π	71	3.89	3.89	11.84	9.68E+05	1.40E+06	0.65E+01
	81	1.34	1.33	7.73	7.77E+07	2.71E+06	1.17E+03
	91	0.74	0.71	5.10	2.33E+07	1.74E+07	3.90E+05
	101	–	0.37	3.20	2.91E+08	2.48E+07	1.64E+07
	111	–	–	1.94	5.11E+07	6.11E+07	2.56E+07

It is well known that the PUFEM suffers from ill-conditioning issues as we increase the number of enrichment functions, for example see [48, 49]. The idea here is to use a small timestep Δt to ensure accuracy while we increase the number of plane wave enrichments until the considered three time integration schemes start to diverge due to the round-off errors and the ill-conditioning issue. The time step $\Delta t = 0.001$ is fixed, and used to simulate a total of 1000 steps to reach the final time of $t = 1$. The same mesh from the previous study is also retained. Table 2 summarizes the numerical accuracy of the three time integration schemes obtained at $t = 1$ for the considered wavenumbers $k = 20\pi, 40\pi$, and 80π.

From the results in Table 2, it is evident that for all the methods, the increase in the number of plane waves leads to a better error and that the higher the wavenumber being solved for, the higher is the Q required to achieve a converged numerical solution. It can also be seen that after a certain number of enrichment functions, the PUFEM-I and PUFEM-E methods fail to produce any good result (entries marked with a '–' in the table), whereas the PUFEM-BD method still produces meaningful results. This suggests that PUFEM-BD method is more stable to increasing Q as compared to PUFEM-I and PUFEM-E methods. To some extent this could be understood with the aid of the condition numbers of the system matrices produced by each of the methods. As can also be seen in Table 2, the condition number related to the matrix produced by PUFEM-BD method is generally lower than that of the other two schemes, especially for lower values of Q. Obviously, increasing Q further will also lead to divergence in the PUFEM-BD method.

To have more insight into the results we also plot in Figure 1 the errors and the associated condition numbers obtained using the considered time stepping schemes for $k = 80\pi$ against increasing values of Q. One can see in this figure that in general PUFEM-E and PUFEM-I methods lead to similar errors and condition numbers but the latter diverges earlier. Although the PUFEM-BD method leads to an order of magnitude higher errors for the same values of Q as the other two methods but the condition numbers associated with PUFEM-BD method are several orders of magnitude smaller. As Q increases beyond a certain threshold i.e. $Q \approx 111$, all three methods are affected with ill conditioning property.

3.1.3 Computational cost

The main advantage of explicit methods is often considered to be their lower computational costs when compared to implicit ones. In this subsection we aim to compare the performance metrics of a computer central processing unit (CPU) with PUFEM-E, PUFEM-BD and PUFEM-I methods.

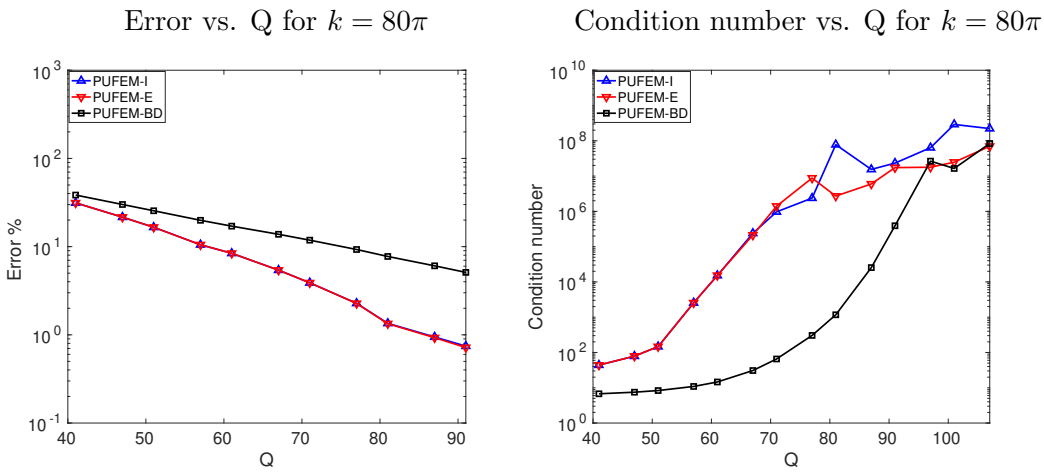

Figure 1: The q-convergence plots (left) and condition number plots (right) obtained using the three methods considered for the example of radial wave, for $k = 80\pi$.

Table 3: Different terms evaluated for the assembly of the finite element linear system of equations using explicit or implicit time integration schemes, at the n^{th} iteration in time.

Time iteration	Method	
	Explicit	Implicit
$n = 1$	$\mathbf{M}, \mathbf{K}, \mathbf{M}_\Gamma$	$\mathbf{M}^{\text{Imp}}, \mathbf{M}_\Gamma$
	$(-\mathbf{K} - \mathbf{M}_\Gamma)\,\mathbf{y}_1, \mathbf{f}_\Omega, \mathbf{f}_\Gamma$	$\mathbf{f}_\Omega^{\text{Imp}}, \mathbf{f}_\Gamma$
$n > 1$	$(-\mathbf{K} - \mathbf{M}_\Gamma)\,\mathbf{y}_1, \mathbf{f}_\Omega, \mathbf{f}_\Gamma$	$\mathbf{f}_\Omega^{\text{Imp}}, \mathbf{f}_\Gamma$

All the methods are implemented by the authors as a sequentially running code in Fortran. The computations are performed on a cluster running CentOS Linux 7 (Core) with an AMD Opteron$^{\text{TM}}$ CPU 6348 @ 1400 MHz $\times$ 24 and 264 GB of RAM. The computational cost, for each iteration in time, is divided into the CPU time spent on the assembly of the linear system and the solution of the linear system. The assembly time is defined as the time taken by the CPU to compute the system matrices and vectors, say matrix $\mathcal{A}$ and vector $\mathbf{b}$ of the linear system of equations $\mathcal{A}\mathbf{x} = \mathbf{b}$. The solution time is the time taken by the CPU to solve this linear system i.e. to obtain the vector $\mathbf{x}$. Furthermore, the assembly and the solution times are compared in the first timestep separately from the subsequent timesteps. The reason for this breakdown in the analysis is to consider the fact that the first iteration in time takes the longest amount of CPU time to complete, for all the three methods. This step involves assembly of the mass and stiffness matrices which are saved and recycled for their usage later on. Also, the inversion of matrix $\mathcal{A}$ takes place in the first step only. Thus, both assembly and solution costs are considerably higher for the first time step, as compared to subsequent time steps. Starting from the second step in time onwards, the total CPU time required for the completion of a full iteration in time associated with each of the considered methods is practically constant. In other words, the variance in the total run-time for the n^{th} iteration in time is negligible for $n > 1$, for all the considered methods. Thus, it is sufficient to observe the trends in the first and the second iterations in time to get an overview of the relative performance of the three methods.

Table 3 provides a summary of the different terms evaluated over Ω and Γ in the explicit and implicit methods in order to assemble the system. The terms evaluated on the boundary Γ are very similar for both explicit and implicit methods, and hence do not contribute to any significant differences in their assembly or solution times. The CPU time comparisons are studied for $k = 80\pi$ considering again the same cases displayed in Figure 1. Figure 2 shows a comparison of the CPU times (in seconds)

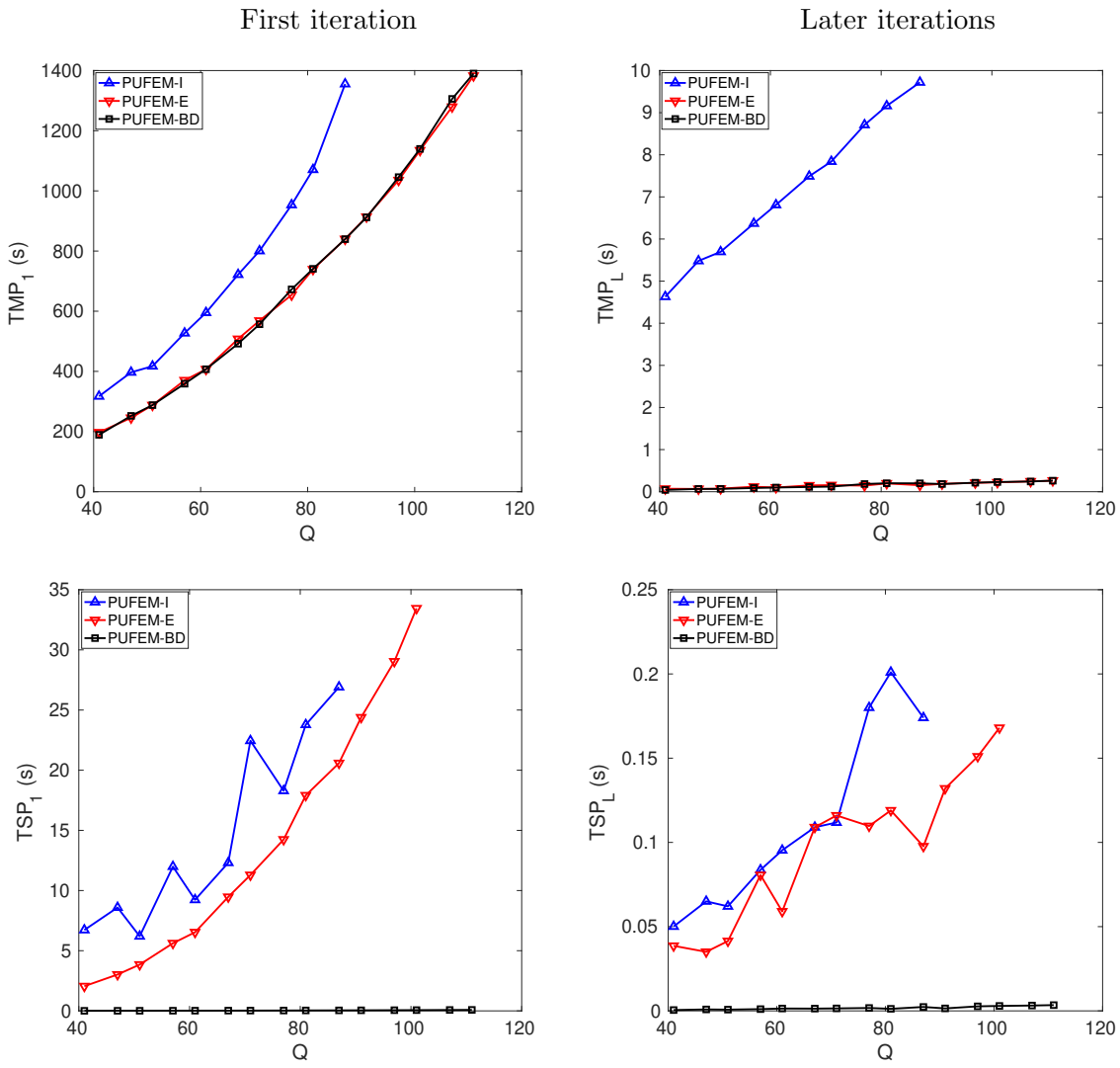

Figure 2: Comparison of the CPU time (in seconds) needed for assembly (top row) and solution (bottom row) for first (left column) and later iterations (right column) against degrees of freedom per node.

against an increased number of enrichment functions Q for each of the three methods i.e. PUFEM-I, PUFEM-E and PUFEM-BD. In this figure, the assembly time required by each method at the first and the later timesteps referred to by TMP_1 and TMP_L, respectively is plotted at the top row whereas the solution times required at the first and later time steps referred to by TSP_1 and TSP_L, respectively at the bottom row.

The results exhibited in Figure 2 show that, using the explicit approach it is possible to significantly reduce the CPU time needed to build the linear system at the first timestep compared to the implicit approach. However, the CPU time for both approaches remains of the same order for the range of Q considered in this study. This can be explained by noting that in the explicit method, the matrices $\mathbf{M}$ and $\mathbf{K}$ are stored in separate variables, whereas in the case of the implicit method, the mass and the stiffness components are added and stored in a single variable ($\mathbf{M}^{\text{Imp}}$). Even though algorithmically, the assembly in the first time step in both implicit and explicit methods requires exactly the same amount of loops, the extra access to memory for storage of two variables instead of one, costs the explicit methods more CPU time. It is noteworthy that, this overhead due to access to memory locations may vary depending on the traffic and type of other processes running concurrently on the CPU node. However, the explicit approach is still intrinsically more efficient because it does

not require the calculation of the solution from the previous two time steps, as is required by the considered implicit method while integrating over Ω to compute $\mathbf{f}_{\mathbf{\Omega}}^{\mathrm{Imp}}$. The equivalent expression computed in case of the explicit approach is $(-\mathbf{K}-\mathbf{M}_{\Gamma})\,\mathbf{y}_1$, which is much faster to evaluate since it does not require spatial integrations, and instead is just a vector to matrix multiplication. From second timestep onward the matrices $\mathbf{M}$ and $\mathbf{K}$ are only reused. Hence, the CPU time for the assembly in the case of explicit approach becomes much faster than implicit. It should be also noted that both PUFEM-E and PUFEM-BD methods require practically similar CPU time for the assembly as they involve evaluating the same terms.
Similarly, we compare the solution processing times required by the three methods against the number of enrichment functions Q. Figure 2 shows the comparisons for the first and the later time steps. As expected, the CPU time for solving the resulting system is practically similar for both PUFEM-I and PUFEM-E methods, since both these methods produce similar system matrices. Obviously with increasing Q the CPU time increases for both PUFEM-I and PUFEM-E methods. On the other hand as delineated from the plots PUFEM-BD provides much better cost effectiveness for increasing Q. For the entire considered range the CPU time remains practically constant at around 0.001 seconds. The reason behind this is the fact that in the case of PUFEM-E and PUFEM-I methods, the size of the inverted matrix is $m^2Q \times m^2Q$ where m^2 is the total number of nodes in the given computational mesh. Whereas, PUFEM-BD method requires the inversion of m^2 non-overlapping blocks of smaller matrices each of size $Q \times Q$, and hence is much faster. It is of interest to note here that the blocks of matrices along the diagonal of the mass matrix in the PUFEM-BD method have no overlap and thus can be inverted simultaneously with distributed computing, for example over different nodes of a cluster, which could further accelerate the performance of the PUFEM-BD method.

3.2 Example of Gaussian pulse

In this section we study the performance of PUFEM-BD method when applied to recover a transient Gaussian pulse. This problem does not include any wavenumbers that could be used to chose the enrichment, which presents a challenge for enrichment approaches. The considered computational domain is $\Omega = [1,2] \times [1,2]$. The analytical solution of the pulse is

$$E(t,x,y) = e^{-a(\beta+r-ct)^2},$$

where $r = \sqrt{x^2+y^2}$ is the radial position, β is the offset that controls the onset of the pulse at the origin, a controls the width of the pulse, and c is the speed at which the pulse propagates. Again the analytical solution is imposed on the wave equation (1) through the initial and boundary conditions. The analyses are carried out with the parameter $\beta = 1$ kept fixed. Three widths are considered, namely, $a = 8, 16$ and 64. The speed of the pulse for all the widths is kept constant at $c = \frac{1}{8\pi}$ with the angular frequency fixed at $\omega = 1$. The simulation is carried out for the time span $t \in]0, 120]$ with the timestep size $\Delta t = 0.001$ where the total number of the computed timesteps is 120000. For the given wave speed this time span allows us to analyse the transit of the pulse through the entire spatial domain. At the start of the simulation, the solution is very close to zero in the entire domain. Evaluating the error as in expression (15) will involve division by numbers close to zero. To remedy this we redefine the relative error as

$$\varepsilon = \frac{\|E_{num} - E\|_{\mathrm{L}^1(\Omega)}}{\|E_{\max}\|_{\mathrm{L}^1(\Omega)}} \times 100\%, \tag{16}$$

where $E_{\max}$ is the magnitude of the exact value of the pulse at its maximum, which is unity in this case.
To evaluate the enrichment wavenumber choice we study the performance of PUFEM-BD method for the Gaussian pulse as we change the pulse width. The impact of the spatial discretization on PUFEM-BD method's accuracy is studied for a range of spatial discretizations. For this study four

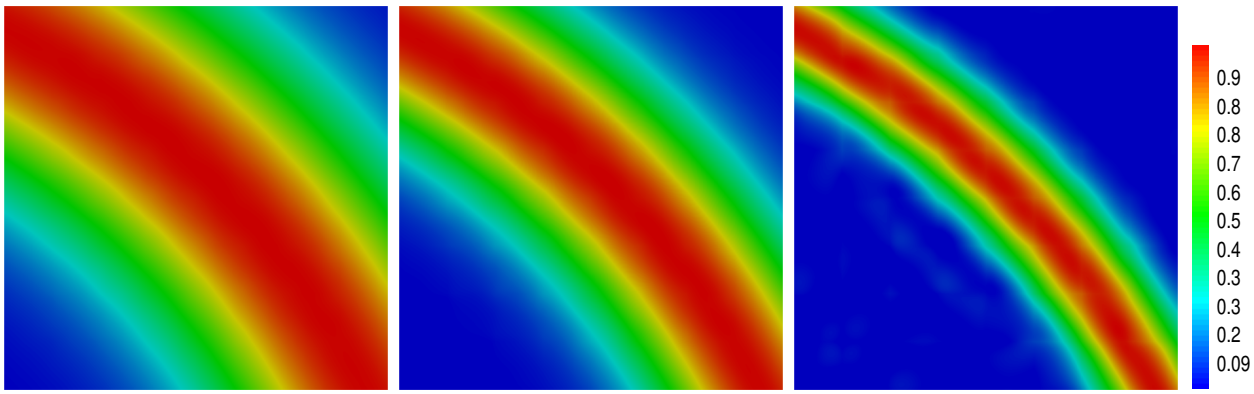

Figure 3: Snapshots of solution obtained with PUFEM-BD method for the example of Gaussian pulse with widths corresponding to $a = 8$, 16 and 64 (left to right) at time $t = 80$. These problems were solved on a uniform mesh with 81 nodes and 64 elements. For each of them the enrichment wavenumber is $k = 4\pi$ and number of enrichment functions is $Q = 9$.

different meshes are used. The meshes are characterised by the chosen number of nodes in one direction $m = \{3, 5, 7, 9\}$. The considered number of enrichment functions are $Q = \{5, 9, 13, 17\}$ and the enrichment wavenumbers are $k = \{2\pi, 4\pi, 8\pi\}$ along with the three different widths corresponding to $a = \{8, 16, 64\}$. The resulting total number of studied cases is 144. The errors for all these cases at the time instance $t = 60$ and at the end of the simulation $t = 120$ are shown in tables 4 and 5, respectively. In these tables the '–' replaces the errors for the cases where the solution diverges. For illustration purposes we show in Figure 3 the solutions obtained with PUFEM-BD method at $t = 80$ for the three considered widths.

Table 4 shows that increasing m for each of the 3 widths will always lead to an improved error. In general for a given m, increasing Q also improves the errors. This improvement seems less significant for $k = 8\pi$. The results in Table 5 again show that increasing m improves the error. For a given m, increasing Q improves the error for the enrichment wavenumber $k = 4\pi$ in all the cases. For the wavenumbers $k = 2\pi$ the error improves for increasing Q for $a = 8$ and 16 while for $a = 64$ the error increases for $m = 3$ by increasing the number of enrichment functions to $Q = 13$. For the case $k = 8\pi$ the error seems to stay practically unchanged for an increasing Q and for all values of a. This suggests that the enrichment has limited contribution to the finite element approximation for $k = 8\pi$ in all considered widths, and for $k = 2\pi$ only in $a = 64$. The FEM approximation in this case is mainly dependent on the element size where only refining the mesh improves the error. Also as the pulse width becomes smaller at $a = 64$ the error becomes larger. This is expected as a finer mesh is needed to capture a narrower pulse.

To understand why the wavenumber $k = 4\pi$ seems to be more suitable for the considered widths, one may link the pulse width to the wavelength of the enrichment functions. For a given value of a, the pulse width could be calculated as $\frac{\sqrt{2\ln 2}}{\sqrt{2a}}$. The related enrichment wavelength will then be twice this number. Thus, for the given values of $a = \{8, 16, 64\}$, the width roughly varies between 0.3, 0.2 and 0.1. These widths will be comparable to the enrichment wavelengths that correspond to the wavenumbers 3.4π, 4.8π and 9.6π. The best results for the given parametric study were found for an enrichment wavenumber of 4π, which is close to the two wavenumbers 3.4π and 4.8π. The errors achieved in this case with $m = 9$ and $Q = 9$ are $\varepsilon = 0.3\%$ and 0.4%. It is well known that the PUFEM suffers from ill-conditioning and stability issues if the characteristic mesh size is not multiple times larger than the enrichment wavelength [17]. These issues become more evident as we increase Q in the sense that the plane waves in the basis are more closely packed together. This explains the instability and the larger errors associated with the PUFEM-BD method, especially observed at higher values of m and smaller values of wavenumber k as we increase Q, as is seen in the tables 4 and 5.

Table 4: The q-refinement with PUFEM-BD method for the example of Gaussian pulse for widths corresponding to $a = 8, 16$, and 64 at time $t = 60$.

		$a=8$				$a=16$				$a=64$			
		m				m				m			
	Q	3	5	7	9	3	5	7	9	3	5	7	9
$k=2\pi$	5	1.31	0.26	–	–	1.30	0.26	–	–	0.83	0.24	–	–
	9	0.55	0.07	–	–	0.59	0.09	–	–	0.41	0.11	–	–
	13	0.44	–	–	–	0.52	–	–	–	0.30	–	–	–
$k=4\pi$	5	4.79	1.07	0.34	0.14	2.84	0.64	0.20	0.08	0.99	0.30	0.11	0.06
	9	3.57	0.40	0.07	0.01	2.17	0.24	0.05	0.02	0.76	0.14	0.04	0.02
	13	3.56	0.38	–	–	2.02	0.23	–	–	0.54	0.12	–	–
$k=8\pi$	5	4.05	1.45	0.71	0.41	3.07	1.29	0.61	0.35	1.33	0.63	0.30	0.13
	9	4.08	1.45	0.52	0.19	3.17	1.19	0.37	0.14	1.40	0.48	0.14	0.05
	13	4.13	1.36	0.50	0.20	3.12	1.10	0.36	0.14	1.34	0.44	0.13	0.05

Our final aim in this example is to evaluate the effect of approximation due to the lumping of the mass matrix by comparing the performance of the PUFEM-BD method to that of the PUFEM-E method. A relatively wide width pulse is considered with $a = 1$. The pulse speed is again fixed at $c = \frac{1}{8\pi}$. The problem is solved on a relatively coarse mesh, namely, $m = 5$ and for an increased Q where the wavenumber of the enrichment plane waves is set as 2π. This choice of k is based on the pulse width which is 0.83 in this case. The timestep size is fixed at $\Delta t = 0.001$. For $Q = 5$ the errors at $t = 60$ are 0.002% and 0.14% with the PUFEM-E method and the PUFEM-BD method, respectively. At the end of the simulations i.e. at $t = 120$, the error increases to 0.004% with the PUFEM-E method and to 1.63% with the PUFEM-BD method. When we increase the number of enrichment functions to $Q = 9$, it is observed that the PUFEM-E method becomes unstable and fails to produce any results whereas the errors with the PUFEM-BD method improve to 0.02% at $t = 60$ and 0.04% at $t = 120$. This result confirms the trends observed in section 3.1.2. The error associated with the PUFEM-BD method is again larger than that of the PUFEM-E method, which is expected because of the approximation due to mass lumping. Also to be seen here is that the PUFEM-BD method is more resilient to increasing Q than the PUFEM-E method. In Figure 4 we plot a comparison of the numerical solutions obtained with the two considered explicit methods, as they evolve in time at a fixed spatial point on the computational domain. For illustration, snapshots of the numerical solution for the pulse recovered with PUFEM-E method for $Q = 5$ at different time instances, are shown in Figure 5.

3.3 Example of transient envelope wave

In this numerical example we recover a transient envelope wave which is described in [38]. The analytical solution of the problem is given as

$$E(t, x, y) = Ae^{\mathrm{i}\omega f_L(t,x,y)},$$

where f_L is a propagator function that controls the initial condition of the problem. Here we use the F_2 propagator, described in details in the reference appendix [38]. The propagator implies a gradual increase in the amplitude of the wave expanding symmetrically around the coordinate origin. Two different wavenumbers, namely, $k = 4\pi$ and $k = 8\pi$ are considered.

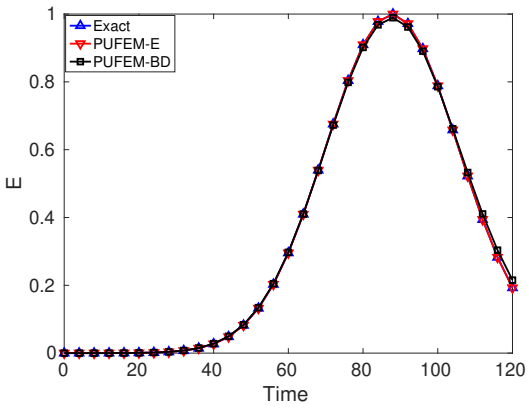

Figure 4: Time evolution of the exact and numerical values of the Gaussian pulse for width $a = 1$, obtained at a spatial point (1.99,1.49) on the computational domain with PUFEM-E and PUFEM-BD methods. Both solutions are obtained over a uniform mesh with 25 nodes and 16 elements, and $Q = 5$ with $k = 2\pi$).

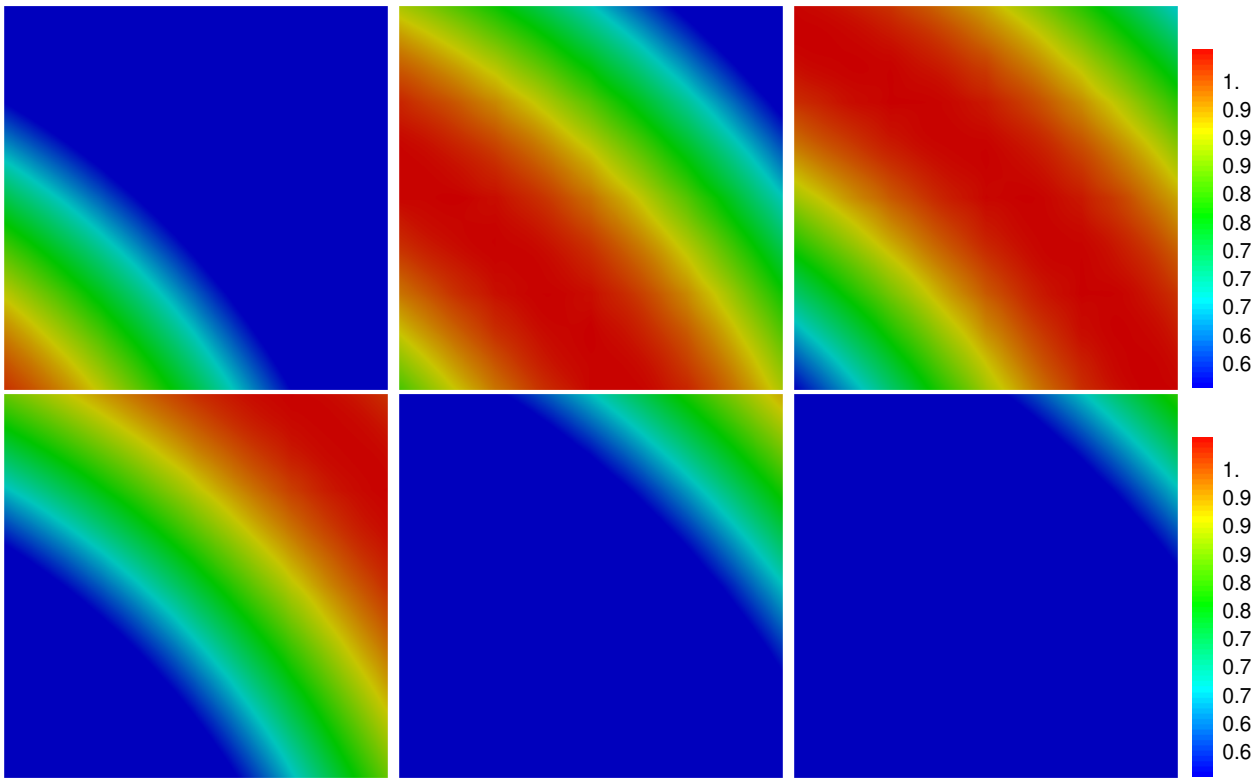

Figure 5: Time evolution of the Gaussian pulse (width corresponding to $a = 1$) in time, solved using PUFEM-E method, for $Q = 5$ and $k = 2\pi$ over a uniform mesh of 16 elements. Snapshots were taken at time $t = 56, 72, 80, 92, 104$, and 108 arranged in the figure from left to right and top to bottom.

Table 5: The q-refinement with PUFEM-BD method for the example of Gaussian pulse for widths corresponding to $a = 8, 16$, and 64 at time $t = 120$.

		$a=8$				$a=16$				$a=64$			
		m				m				m			
	Q	3	5	7	9	3	5	7	9	3	5	7	9
	5	14.9	3.8	–	–	21.3	8.5	–	–	20.3	16.4	–	–
$k = 2\pi$	9	3.8	0.4	–	–	8.9	1.8	–	–	15.5	13.8	–	–
	13	2.6	–	–	–	8.1	–	–	–	22.4	–	–	–
	5	22.2	6.9	10.3	5.6	20.2	12.1	10.3	5.0	16.7	18.0	13.1	8.8
$k = 4\pi$	9	14.3	6.4	1.0	0.3	15.9	8.6	1.7	0.4	17.7	11.0	4.6	2.3
	13	14.2	3.2	–	–	12.7	3.5	–	–	15.6	8.1	–	–
	5	17.5	18.5	12.5	7.1	14.4	20.5	18.0	10.9	9.8	17.4	20.3	16.8
$k = 8\pi$	9	17.8	13.2	5.0	2.2	15.0	16.4	9.4	4.0	11.6	18.9	13.7	7.3
	13	16.8	17.5	4.8	3.8	14.2	21.2	6.3	7.9	11.9	22.2	6.3	10.7

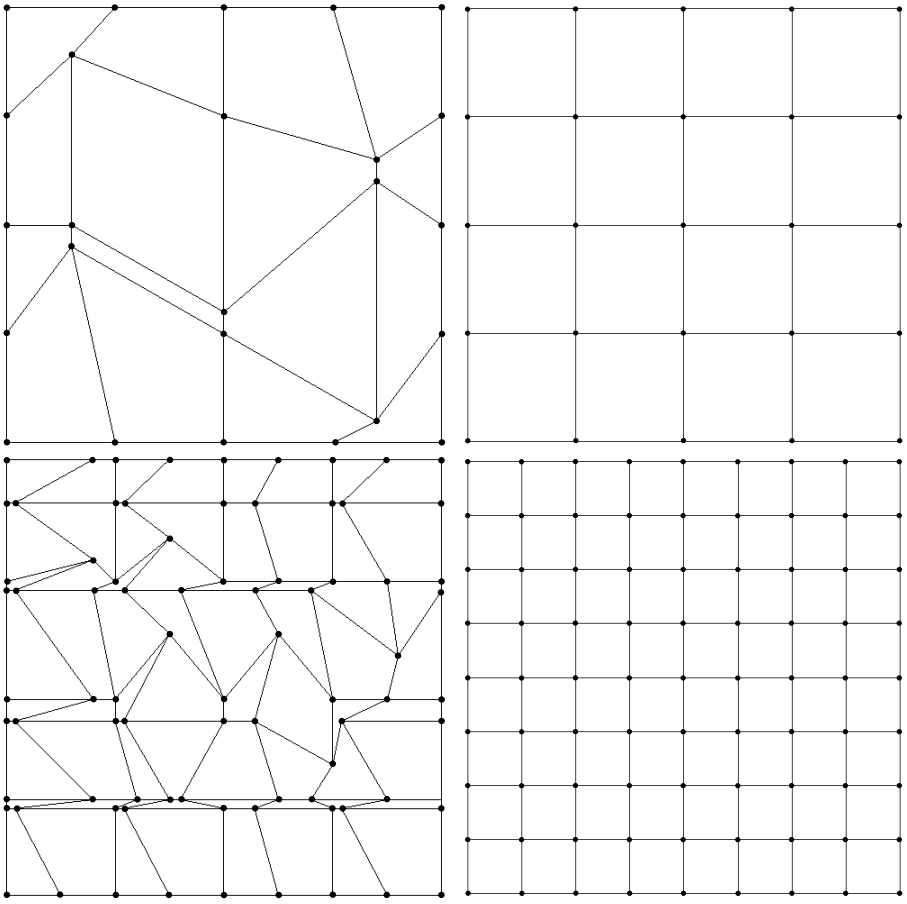

Figure 6: Uniform meshes and their distorted counterparts, used in the example of transient envelope wave.

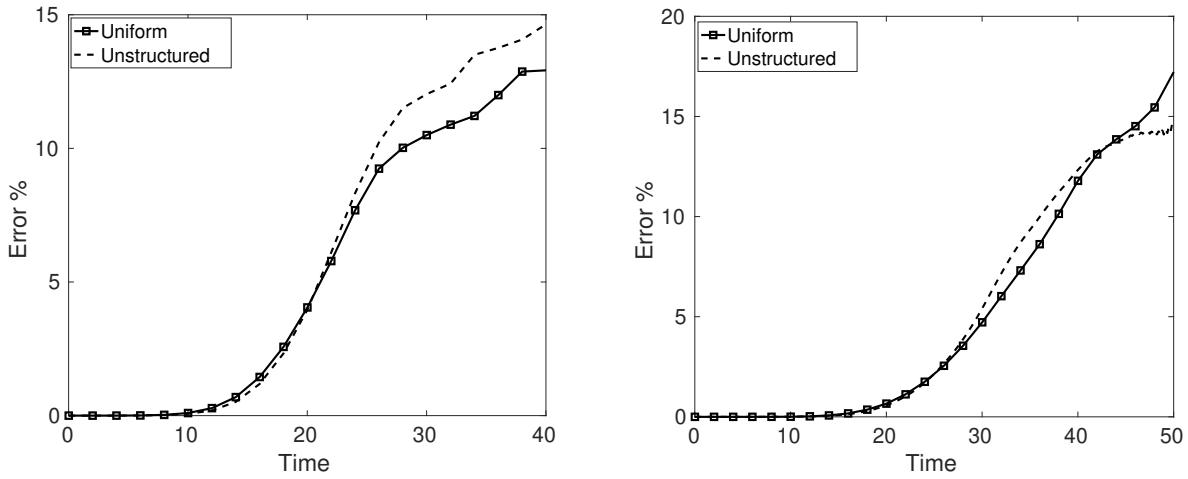

Figure 7: Comparison of errors associated with numerical solutions for the example of transient envelope wave, obtained using the PUFEM-BD method over uniform and unstructured meshes respectively. The plots are for $k = 4\pi$ (left) and $k = 8\pi$ (right), for $Q = 13$ and $Q = 17$, respectively.

The aim in this example is to evaluate the impact of using an unstructured mesh on the approximation levied upon by the lumped mass approach. To this end we consider two sets of meshes, first with uniform elements similar to the meshes considered in the previous studies, and second with distorted elements. The unstructured meshes of the distorted elements are obtained by randomly displacing the nodes of the uniform meshes. It should be stressed that only the locations of the nodes, are displaced so that the distorted meshes include the same number of nodes and elements as their uniform counterparts but the elements are no longer square in shape. Furthermore, the distortion of the meshes is performed to produce severely poor quality elements. Figure 6 shows the two uniform meshes considered in this study alongside the unstructured meshes. The coarse meshes composed of 25 nodes and 16 elements, are then enriched using $Q = 13$ to solve the problem for $k = 4\pi$. Similarly the fine meshes composed of 81 nodes and 64 elements are enriched with $Q = 17$ and used to solve for $k = 8\pi$. The step-size in time in both the cases is kept constant at $\Delta t = 0.01$. The simulation is completed at $t = 40$ for $k = 4\pi$ and $t = 50$ for $k = 8\pi$, respectively, allowing the waves to spread into the entire computational domain. It is noteworthy that the angular frequency in time is kept constant at 1, thus for a higher wavenumber, the wave moves slower. The time evolution of the errors associated with the PUFEM-BD method for the considered two wavenumbers are then displayed in Figure 7. Figures 8 and 9 show the numerical solutions obtained with the PUFEM-BD method for the two considered wavenumbers, respectively, at different time instances. The results are very similar to that obtained with the reference implicit method in [38].

The results in Figure 7 show that the errors are relatively similar if using a structured or an unstructured mesh for the start of the simulation. For example for $k = 4\pi$ both meshes lead to similar errors up to time $t \approx 19$, while for $k = 8\pi$ up to time $t \approx 25$. Thereafter, the unstructured meshes cause a significant deviation in the error evolution. To compare this behavior to the explicit time integration scheme without lumping, the problem is solved again using the PUFEM-E method on the same meshes shown in Figure 6. On the uniform meshes, the errors in numerical solution at the end of the simulation with the PUFEM-E method are 5.97% for $k = 4\pi$ on the coarse uniform mesh with $Q = 7$, and is 7.64% for $k = 8\pi$ on the fine uniform mesh with $Q = 11$. These errors are compared to 12.91% for $k = 4\pi$ with $Q = 13$ and 17.21% for $k = 8\pi$ with $Q = 17$, using the PUFEM-BD method. However if we increase Q any further, the PUFEM-E method becomes unstable for both the wavenumbers considered. Moreover, the PUFEM-E method fails to produce any results when distorted meshes are used. This shows that the lumping approach is more stable not only for an increased Q, as is also

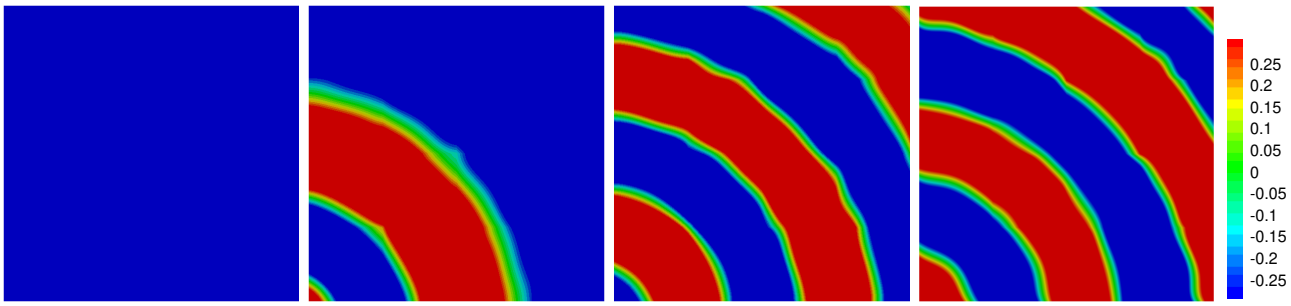

Figure 8: Snapshots of numerical solutions obtained with the PUFEM-BD method for the example of transient envelope wave, solved on a uniform grid with 16 elements and $Q = 13$ with $k = 4\pi$. These snapshots correspond to time t=10, 20, 30 and 40 (left to right).

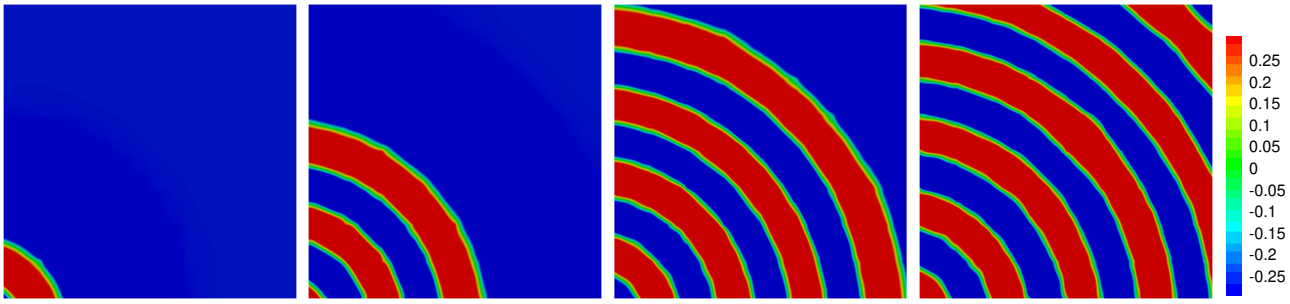

Figure 9: Snapshots of numerical solutions obtained with the PUFEM-BD method for the example of transient envelope wave, solved on a uniform grid with 64 elements and $Q = 17$ with $k = 8\pi$. These snapshots correspond to time t=20, 30, 40 and 50 (left to right).

seen in the previous test cases, but also is more stable when using unstructured meshes.

3.4 Example of resonant cavity

In this final numerical example we study the solution of the wave equation subjected to homogeneous boundary and initial conditions, with a non-zero source inside the domain. Such problems arise in many real world applications, such as in electromagnetic compatibility analysis and modal analysis for waveguides [3,12]. In particular, we consider two different areas of applications, as shown in figures 10 and 13. In the first application we consider a waveguide that is typically used inside a microwave oven to illuminate the oven chamber with electromagnetic radiation [50]. The four walls of the waveguide create a resonant cavity that reflects the inbound radiation inside it. The second application is for a circular cross-section of a high voltage insulated steel armoured electrical cable. The steel armour, that is used to provide sustenance for the high voltage electrical cables to endure mechanical stress as they are buried underground, also traps electromagnetic disturbances occurring inside the cable due to current imbalances and material defects [51]. The solution of the wave equation inside these geometries provides the distribution of the electromagnetic energy within the material, which then could be coupled with the diffusion equation to analyse various thermal properties [52, 53]. Owing to the increase in power demands both for industrial and household usage, such analyses are deemed useful to evaluate performance and prevent ageing of cables. For the purpose of this paper, we define

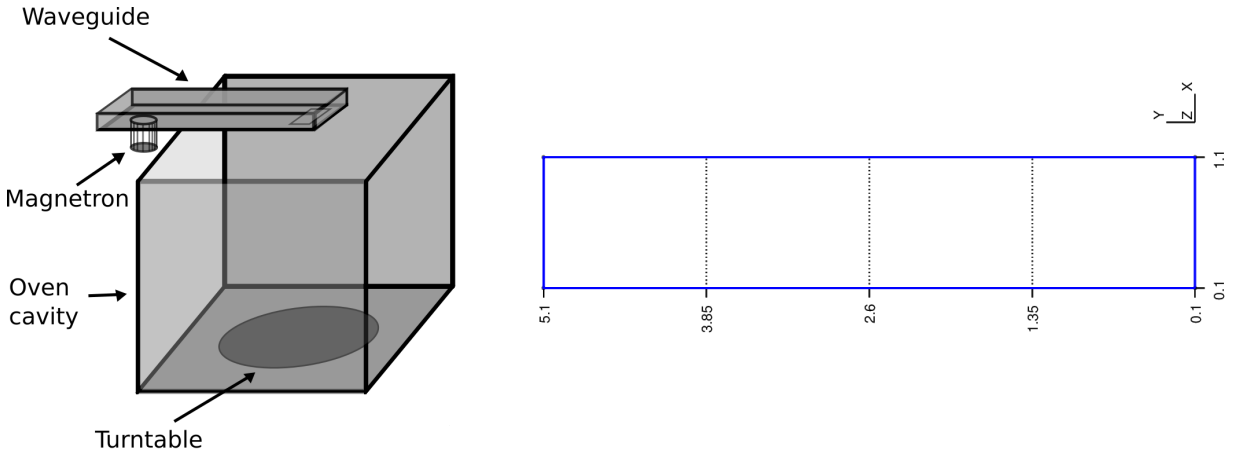

Figure 10: Rectangular waveguide inside a microwave oven (left) and modelled waveguide geometry used for analysis (right).

a model problem as

$$\frac{1}{c^2}\frac{\partial^2 E}{\partial t^2} - \nabla^2 E = f(t,\mathbf{x}), \qquad (t,\mathbf{x}) \in [0,T[\times\Omega, \tag{17a}$$

$$E(t,\mathbf{x}) = 0, \qquad (t,\mathbf{x}) \in [0,T[\times\Gamma, \tag{17b}$$

$$E(0,\mathbf{x}) = 0, \qquad \mathbf{x} \in \Omega, \tag{17c}$$

$$\frac{\partial E}{\partial t}(0,\mathbf{x}) = 0, \qquad \mathbf{x} \in \Omega. \tag{17d}$$

To solve for the scalar field value E inside the domain Ω for a given source term $f(t,\mathbf{x})$, the domain and the source term are defined separately for the two considered geometries. Since the solution vanishes on the boundary, the weak form for this example is given as

$$\int_{\Omega'} \frac{\partial E}{\partial t}\phi \, d\Omega = \int_{\Omega'} V\phi \, d\Omega, \tag{18a}$$

$$\frac{1}{c^2}\int_{\Omega'} \frac{\partial V}{\partial t}\phi \, d\Omega = -\int_{\Omega'} \nabla E \cdot \nabla\phi \, d\Omega + \int_{\Omega'} f(t,\mathbf{x})\phi \, d\Omega, \tag{18b}$$

where $\Omega' = \{\mathbf{x} : \mathbf{x} \in \Omega \text{ and } \mathbf{x} \notin \Gamma\}$. Notice that now we do not have any matrices or vectors to be computed over the boundary, and thus we do not need to evaluate $\mathbf{M}_\Gamma$ and $\mathbf{f}_\Gamma$, as we did for previous examples. Also, there's no need to solve the extra Galerkin equations of (11) due to the homogeneous initial conditions.

3.4.1 Rectangular waveguide

For the waveguide geometry, as seen in Figure 10, we consider

$$f(t,x,y) = \sin\left(\frac{(x-a)\pi}{0.2}\right)\sin\left(\frac{(y-a)\pi}{0.2}\right)\sin(\pi t), \tag{19}$$

where $(x,y) \in [a, a+0.2] \times [a, a+0.2]$ and $a = 0.55$. The considered computational domain is $\Omega = [0.1, 1.1] \times [0.1, 5.1]$, and the problem is solved for the total time period $T = 10$. The step-size in time is $\Delta t = 0.001$ and the simulation is run for a total of 10000 time steps. The wave speed $c = \frac{1}{\pi}$. To arrive at the converged solution, we adopt the same approach of mesh refinement and increasing number of nodal enrichment plane waves, as used in subsection 3.2 for the example of a Gaussian pulse. Since for this example the angular frequency, as seen from equation (19), is $\omega = \pi$ we use

$k = \omega/c = \pi^2$ as the wavenumber for the plane wave enrichments. We consider three uniform meshes, with rectangular elements, consisting of 45, 171, and 741 nodes respectively, termed here as meshes m1, m2, and m3. Subsequently for each mesh, we consider three different number of enrichment functions $Q = \{5, 7, 9\}$ and solve the wave problem using the PUFEM-BD method, to get a total of 9 different numerical solutions. These numerical solutions are termed as BD-miQj, where $i \in \{1, 2, 3\}$ denotes the mesh number and $j \in \{5, 7, 9\}$ denotes the number of enrichment functions used per node. The total degrees of freedom associated with these BD-miQj solutions ranges from 225 (for BD-m1Q5) to 6669 (for BD-m3Q9). For comparison, we also solve the wave problem on a very fine mesh with a total of 12561 nodes using the first order FEM, without using any nodal enrichment. Figure 11 compares these BD-miQj numerical solutions against the solution obtained with the non-enriched FEM. In this figure we plot the time evolution of the numerical solution recorded at the point (x=0.6,y=2.6) inside the domain Ω. It can be clearly seen that as we increase the total degrees of freedom used with the PUFEM-BD method, the solution converges towards the reference FEM solution. Here we define the percentage error as

$$\varepsilon = \frac{\|E_{num} - E_{FEM}\|_{\mathrm{L}^1(\Omega)}}{\|E_{\max}\|_{\mathrm{L}^1(\Omega)}} \times 100\%, \tag{20}$$

where E_{FEM} is the reference solution obtained with the non enriched FEM over a very fine mesh, and $E_{\max}$ is the magnitude of the source term at its maximum, which is unity in this case. The percentage error is plotted for individual BD-miQj solutions, recorded at the reference point (0.6,2.6), as shown in Figure 11. It is observed from this figure that the relative error associated with the three BD-m1Qj solutions is below 6%, and with the BD-m2Qj and BD-m3Qj solutions is below 1%, throughout the total duration of the simulation. We also test the performance of the PUFEM-BD method for this geometry over an unstructured mesh, as shown in Figure 12. The unstructured mesh consists of 655 nodes, and was obtained using the open-source package GMSH. For this mesh, the number of enrichment functions $Q = 9$ was used. This resulted into a total of 5895 degrees of freedom, which is comparable to that associated with the BD-m3Q9 method. The numerical results obtained with the unstructured mesh are also shown in Figure 12, along with the BD-m3Q9 solution and the reference FEM solution.

3.4.2 High voltage cable cross section

For the circular cross-section geometry, we consider

$$f(t, x, y) = \sin\left(\frac{(x+1)\pi}{2}\right) \sin\left(\frac{(y+9)\pi}{1}\right) \sin(t), \tag{21}$$

for $(x, y) \in [-1, 1] \times [-9, -8]$. In the modelled geometry, the outer and the inner circles form the boundary Γ where the solution vanishes. The radii of the outer and the inner circles are 12 and 4 units respectively. The centre of the outer circle corresponds to the origin. The centres of the inner circles are equidistant from the origin, located at the points (0,6), (-5.19,-3), and (5.19,-3) that form an equilateral triangle. The considered computational domain is shown in Figure 13. The problem is solved for the total time period $T = 70$. The step-size in time is $\Delta t = 0.001$ and the simulation is run for a total of 70000 time steps. The wave speed $c = \frac{1}{\pi}$. The angular frequency $\omega = 1$ for the source term as seen in equation (21), which provides the wavenumber $k = \pi$ to be used for the enrichment functions. To arrive at a converged solution, we again employ the aforementioned strategy of q-convergence and mesh refinement. To this end we consider six different meshes with a total of 364, 588, 962, 1336, 1842, and 2342 nodes respectively, termed here as meshes m1, m2, m3, m4, m5, and m6. Subsequently for each mesh, we consider three different number of enrichment functions $Q = \{5, 7, 9\}$ and solve the wave problem of equation (17) using the PUFEM-BD method. This gives us 18 different numerical solutions, termed here as BD-miQj using the same convention as for the numerical solutions for the waveguide geometry. Here, the total number of degrees of freedom ranges

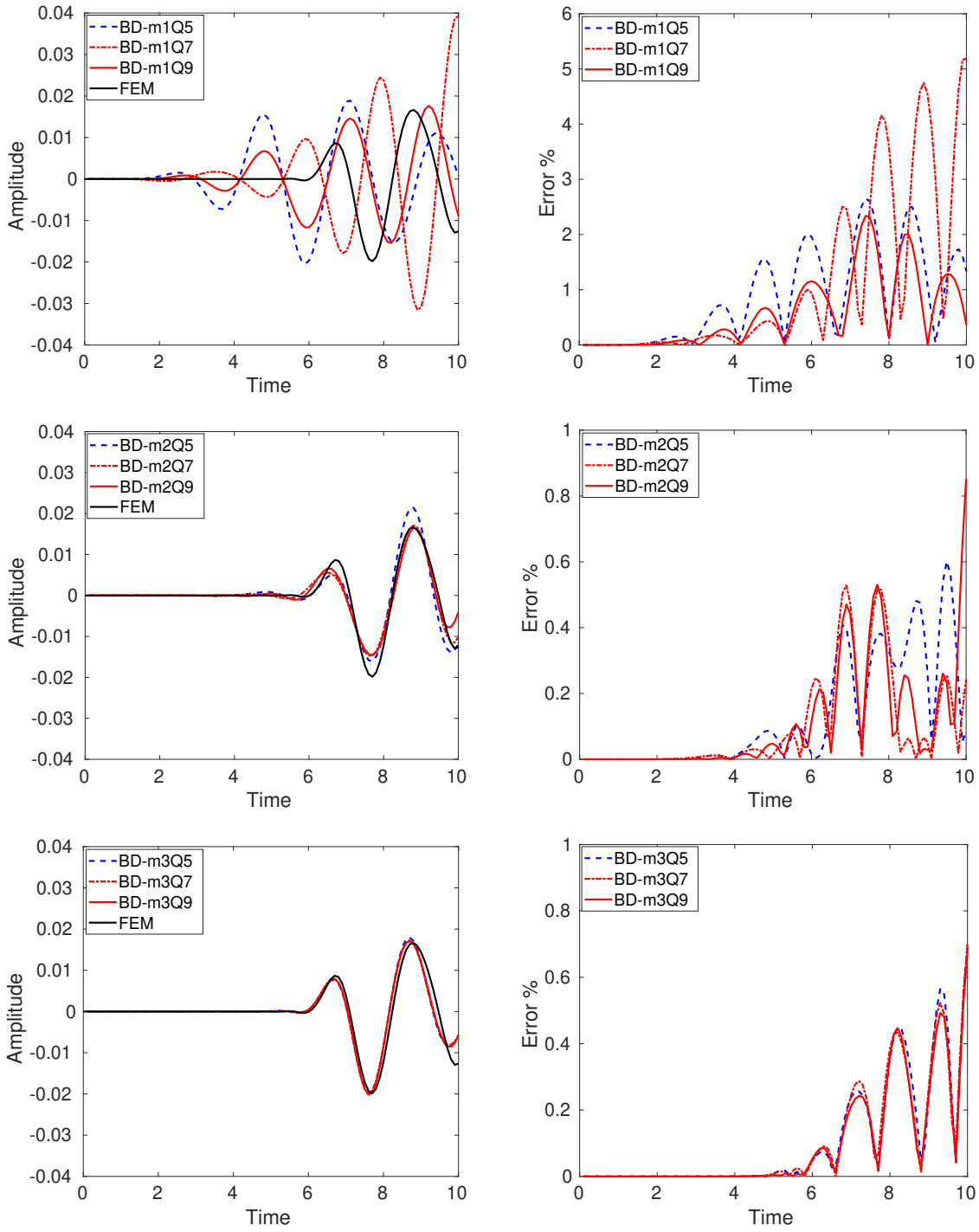

Figure 11: Comparison of the numerical solutions obtained with the PUFEM-BD method against the reference FEM solution, for the rectangular waveguide example. The graphs show (left column) the time evolution of the amplitude of the scalar field E at a point (0.6,2.6) inside the computational domain Ω, and (right column) the relative error in these amplitudes obtained with the PUFEM-BD solutions as compared to the fine mesh FEM solution.

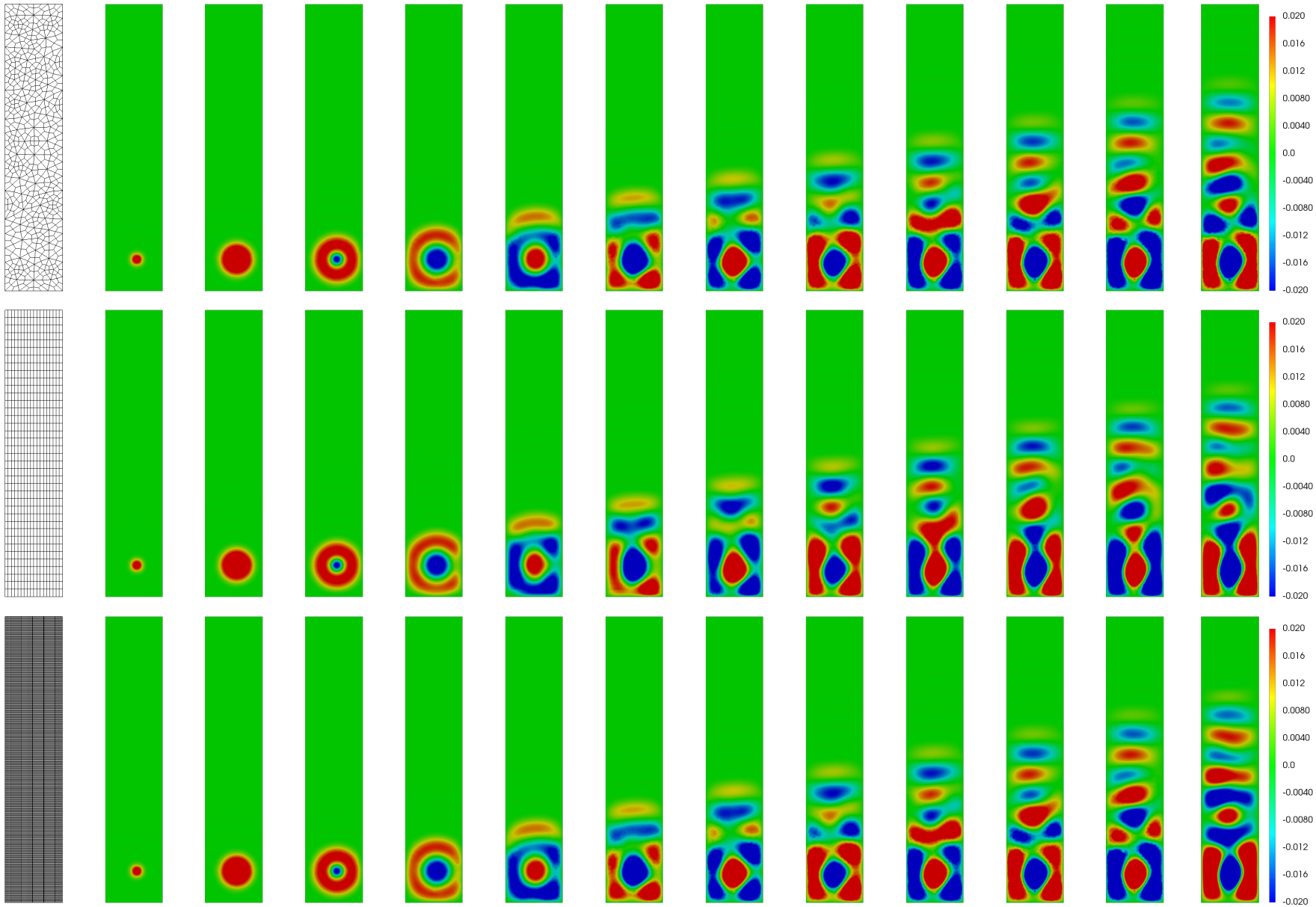

Figure 12: Computational mesh (far left), and the snapshots of the numerical solutions obtained with the PUFEM-BD method for the example of rectangular waveguide. Solutions are for (top) the unstructured grid with 655 nodes and $Q = 9$, (middle) the structured mesh m3 with 741 nodes and $Q = 9$. Reference solution (bottom) is obtained with linear FEM on the structured very fine mesh with 12561 nodes. These snapshots correspond to the time instants t =0.4, 1.1, 1.5, 2, 3, 4, 5, 6, 7, 8, 9 and 10 (left to right). The range on the colour legend is (-0.02,0.02).

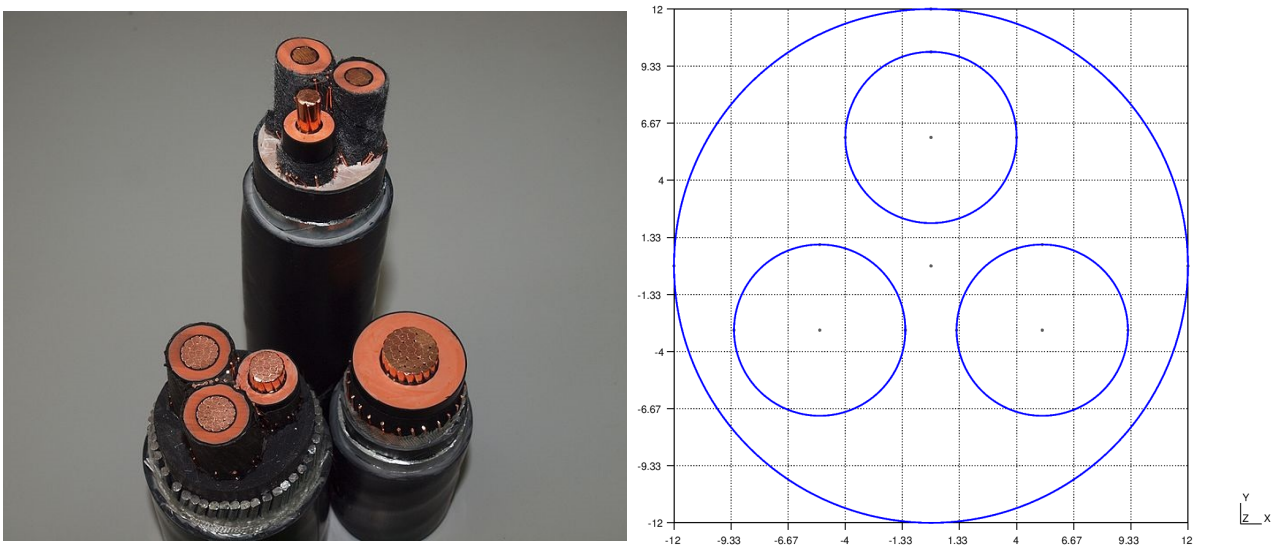

Figure 13: High voltage electrical cables (left) and modelled geometry for the circular cross-section of an electrical cable (right).

from 1820 (for BD-m1Q5) to 21078 (for BD-m6Q9). The time evolution for the numerical solutions BD-m4Q9, BD-m5Q9 and BD-m6Q9 are shown in Figure 14, depicting mesh refinement for $Q = 9$ to obtain a converged solution.
From the test cases studied in this section, it can be clearly seen that the numerical solutions converge for the PUFEM-BD method with increasing degrees of freedom as expected. The method is also seen to be applicable with unstructured meshes, and produces similar results when compared with solutions obtained using structured meshes. The PUFEM-BD method could also prove very useful to reduce the time complexities associated with solving the numerical problem, by using parallel computing to distribute the inversion of non-overlapping blocks of the system matrix using concurrent computational tools such as OpenMP. This feature could be very useful especially when solving real world problems with massive geometries.

4 Conclusions

In this paper we propose using explicit time integration scheme with block diagonal lumping of the mass matrix for the partition of unity finite element solution of time domain wave problems. We study the convergence of the solution for refined temporal and spatial discretizations. The results obtained with the explicit scheme both lumped and full, are compared to the results obtained with an implicit time integration scheme. The paper shows major advantages for the explicit approach with a lumped mass matrix. The advantages are related to significant reduction in the CPU time as well as better conditioned linear system of equations.
The work also investigates the partition of unity solution of a transient Gaussian pulse which does not involve a specific wavenumber. A parametric study shows that the plane wave enrichment can still be used for the solution. Better errors are achieved for the cases where the wavelength of the enrichment functions is close to the pulse width. More rigorous methods could be developed for isolating the spatial frequencies present in the solution, albeit that would form the basis of future work in developing the techniques presented in this paper.
Inherently, the explicit time integration schemes, as opposed to their implicit counterparts, are only conditionally stable. The CFL condition must be satisfied to ensure their numerical stability. This condition essentially requires that the problem is solved numerically on a fine enough discretization in both space and time, such that the numerical solver is able to capture the fastest of the wave

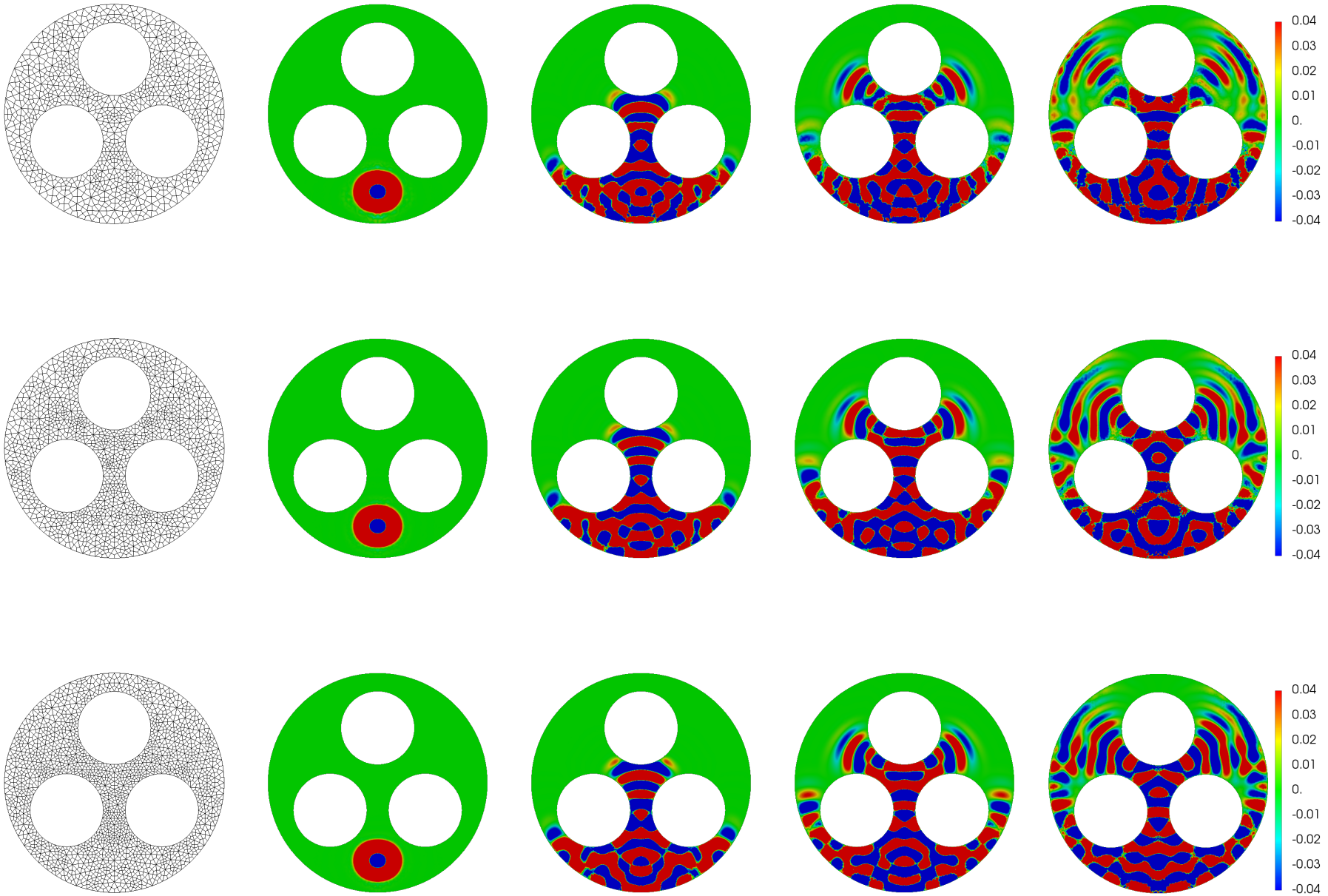

Figure 14: Computational mesh (far left), and the snapshots of the numerical solutions obtained with the PUFEM-BD method for the example of high voltage cable cross section. Solutions shown are for (top) the unstructured mesh m4 with 1336 nodes using $Q = 9$, (middle) mesh m5 with 1842 nodes using $Q = 9$, and (bottom) mesh m6 with 2342 nodes using $Q = 9$. These snapshots correspond to the time instants $t = 6.8$, 37.2, 53.2, and 70 (left to right). The range on the colour legend is (-0.04,0.04).

components present in the actual solution. The oscillatory nature of the enrichment functions used to improve the finite element basis allows us to use significantly coarse meshes even for high wavenumbers. The coarse spatial mesh in turn facilitates utilizing the benefits of an explicit scheme without the need for a fine timestep size. The discussed numerical results show that the timestep size used with the considered implicit scheme is very similar to that used with the proposed explicit counterparts. The PUFEM is known to produce ill-conditioned linear system of equations with an increased nodal density of the enrichment functions. The results discussed in this paper show that the lumped mass approach improves the conditioning of the PUFEM system matrix by several orders of magnitude. Furthermore, the resulting block diagonal system matrix is much faster to invert, and can be stored much more efficiently as compared to the full matrix for the non-lumped approach. The results also suggest that lumping the mass matrix makes the PUFEM more stable against poor quality distorted elements used to discretize the spatial mesh. Although the block diagonal lumping of the mass matrix is only tested for the partition of unity method, and in the context of wave propagation problems, the approach could also be used for other enrichment techniques such as the generalized finite element method when solving time-domain problems.

Acknowledgements

The first author wishes to thank Heriot Watt University for providing support through a fee waiver scholarship. The second author is partially supported by Japan Society for the Promotion of Science Invitational Fellowship L19554. The geometries in the subsection 3.4 are modelled using the open-source package GMSH.

References

[1] C.L. Bennett, G.F. Ross, Time-domain electromagnetics and its applications, Proc. IEEE 66 (1978) 299–3182.

[2] T. Tamir, G. Griffel, H.L. Bertoni, Guided-wave optoelectronics, Springer, Boston, 1995.

[3] J.F. Lee, Finite element methods for microwave engineering, in: J.F. Kiang (Ed.), Novel technologies for microwave and millimeter–wave applications, Springer, 2004, pp. 285–301.

[4] D.B. Davidson, Computational electromagnetics for RF and microwave engineering, Cambridge University Press, 2011.

[5] K. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE Trans. Antennas Propag. 14 (1966) 302–307.

[6] A. Taflove, S.C. Hagness, Computational electrodynamics: The finite-difference time-domain method, third ed., Artech House, 2005.

[7] A. Taflove, M.E. Brodwin, Numerical solution of steady-state electromagnetic scattering problems using the time-dependent Maxwell's equations, IEEE Trans. Micro. Theory Tech. 23 (1975) 623–630.

[8] A. Buffa, M. Costabel, C. Schwab, Boundary element methods for Maxwell's equations on non-smooth domains, Numer. Math. 92 (2002) 679–710.

[9] A. Buffa, R. Hiptmair, Galerkin boundary element methods for electromagnetic scattering, in: M. Ainsworth, P. Davies, D. Duncan, B. Rynne, P. Martin (Eds.), Topics in computational wave propagation. Lecture Notes in Computational Science and Engineering, Springer, 2003, pp. 83–124.

[10] A. Buffa, R. Hiptmair, T. Petersdorff, C. Schwab, Boundary element methods for Maxwell transmission problems in Lipschitz domains, Numer. Math. 95 (2003) 459–485.

[11] J.M. Jin, The finite element method in electromagnetics, Wiley, 2014.

[12] D.M. Pozar, Microwave engineering, Wiley, 2004.

[13] J.M. Dlugach, M.I. Mishchenko, L. Liu, D.W. Mackowski, Numerically exact computer simulations of light scattering by densely packed, random particulate media, J. Quant. Spectrosc. Radiat. Transf. 112 (2011) 2068–2078.

[14] H. Wu, Z.L. Li, Scale issues in remote sensing: A review on analysis, processing and modeling, Sens. 9 (2009) 1768–1793.

[15] J.M. Melenk, I. Babuška, The partition of unity finite element method: Basic theory and applications, Comput. Methods Appl. Mech. Eng. 139 (1996) 289–314.

[16] O. Laghrouche, P. Bettess, R.J. Astley, Modelling of short wave diffraction problems using approximating systems of plane waves, Int. J. Numer. Methods Eng. 54 (2002) 1501–1533.

[17] K. Christodoulou, O. Laghrouche, M.S. Mohamed, J. Trevelyan, High-order finite elements for the solution of helmholtz problems, Comput. Struct. 191 (2017) 129–139.

[18] O. Laghrouche, P. Bettess, E. Perrey-Debain, J. Trevelyan, Wave interpolation finite elements for Helmholtz problems with jumps in the wave speed, Comput. Methods Appl. Mech. Eng. 194 (2005) 367–381.

[19] M.S. Mohamed, O. Laghrouche, A.El. Kacimi, Some numerical aspects of the PUFEM for efficient solution of 2D Helmholtz problems, Comput. Struct. 88 (2010) 1484–1491.

[20] T. Strouboulis, I. Babuška, R. Hidajat, The generalized finite element method for Helmholtz equation: Theory, computation, and open problems, Comput. Methods Appl. Mech. Eng. 195 (2006) 4711–4731.

[21] T. Strouboulis, R. Hidajat, I. Babuška, The generalized finite element method for Helmholtz equation part II: Effect of choice of handbook functions, error due to absorbing boundary conditions and its assessment, Comput. Methods Appl. Mech. Eng. 197 (2008) 364–380.

[22] T. Huttunen, P. Monk, F. Collino, J.P. Kaipio, The ultra weak variational formulation for elastic wave problems, Soc. Ind. Appl. Math. J. Sci. Comput. 25 (2004) 1717–1742.

[23] T. Luostari, T. Huttunen, P. Monk, The ultra weak variational formulation using bessel basis functions, Commun. Comput. Phys. 11 (2012) 400–414.

[24] C. Farhat, I. Harari, U. Hetmaniuk, A discontinuous Galerkin method with Lagrange multipliers for the solution of Helmholtz problems in the mid-frequency regime, Comput. Methods Appl. Mech. Eng. 192 (2003) 1389–1419.

[25] S. Petersen, C. Farhat, R. Tezaur, A space–time discontinuous Galerkin method for the solution of the wave equation in the time domain, Int. J. Numer. Methods Eng. 78 (2009) 275–295.

[26] D. Wang, R. Tezaur, C. Farhat, A hybrid discontinuous in space and time Galerkin method for wave propagation problems,Int. J. Numer. Methods Eng. 99 (2014) 263–289.

[27] R. Griesmaier, P. Monk, C. Farhat, Discretization of the wave equation using continuous elements in time and a hybridizable discontinuous Galerkin method in space, J. Sci. Comput. 58 (2014) 472–498.

[28] F. Kretzschmar, A. Moiola, I. Perugia, S. M. Schnepp, A priori error analysis of spacetime Trefftz discontinuous Galerkin methods for wave problems, IMA J. Numer. Anal. 36 (2015) 1599–1635.

[29] G.C. Diwan, M.S. Mohamed, M. Seaid, J. Trevelyan, O. Laghrouche, Mixed enrichment for the finite element method in heterogeneous media, Int. J. Numer. Methods Eng. 101 (2015) 54–78.

[30] A.El. Kacimi, O. Laghrouche, Numerical modelling of elastic wave scattering in frequency domain by the partition of unity finite element method, Int. J. Numer. Methods Eng. 77 (2018) 1646–1669.

[31] M.S. Mahmood, O. Laghrouche, J. Trevelyan, A.El. Kacimi, Implementation and computational aspects of a 3D elastic wave modelling by PUFEM, Appl. Math. Model. 49 (2017) 568–586.

[32] J. Jiang, M.S. Mohamed, M. Seaid, H. Li, Identifying the wavenumber for the inverse Helmholtz problem using an enriched finite element formulation, Comput. Methods Appl. Mech. Eng. 340 (2018) 615–629.

[33] M.S. Mohamed, M. Seaid, J. Trevelyan, O. Laghrouche, A partition of unity fem for time-dependent diffusion problems using multiple enrichment functions, Int. J. Numer. Methods Eng. 93 (2013) 245–265.

[34] M.S. Mohamed, M. Seaid, J. Trevelyan, O. Laghrouche, An enriched finite element model with q-refinement for radiative boundary layers in glass cooling, J. Comput. Phys. 258 (2014) 718–737.

[35] M.S. Mohamed, M. Seaid, J. Trevelyan, O. Laghrouche, Time-independent hybrid enrichment for finite element solution of transient conductionradiation in diffusive grey media, J. Comput. Phys. 251 (2013) 81–101.

[36] R. Hiptmair, A. Moiola, I. Perugia, A survey of Trefftz methods for the Helmholtz equation, in: G. Barrenechea, F. Brezzi, A. Cangiani, E. Georgoulis (Eds.), Building Bridges: Connections and Challenges in Modern Approaches to Numerical Partial Differential Equations. Lecture Notes in Computational Science and Engineering, Springer, 2016, pp. 237–279.

[37] S. Ham, K.J. Bathe, A finite element method enriched for wave propagation problems, Comput. Struct. 94-95 (2012) 1–12.

[38] M. Drolia, M.S. Mohamed, O. Laghrouche, M. Seaid, J. Trevelyan, Enriched finite elements for initial-value problem of transverse electromagnetic waves in time domain, Comput. Struct. 182 (2017) 354–367.

[39] Y. Yang, H. Zheng, M.V. Sivaselvan, A rigorous and unified mass lumping scheme for higher-order element, Comput. Methods Appl. Mech. Eng. 319 (2017) 491–514.

[40] Y. Yang, D. Xu, H. Zheng, Explicit discontinuous deformation analysis method with lumped mass matrix for highly discrete block system, Int. J. Geomech. 18 (2018) 04018098.

[41] J. Chan, J.A. Evans, Multi-patch discontinuous Galerkin isogeometric analysis for wave propagation: Explicit time-stepping and efficient mass matrix inversion, Comput. Methods Appl. Mech. Eng. 333 (2018) 22–54.

[42] G. Cohen, Higher-order numerical methods for transient wave equations, Springer, 2002.

[43] T. Hughes, The finite element method: Linear static and dynamic finite element analysis, Dover, 2000.

[44] O. Zienkiewicz, R. Taylor, J.Z. Zhu, The finite element method: Its basis and fundamentals, seventh ed., Butterworth-Heinemann, 2013.

[45] M.A. Schweitzer, Variational mass lumping in the partition of unity method, Soc. Ind. Appl. Math. 35 (2013) 1073–1097.

[46] A.El. Kacimi, O. Laghrouche, Improvement of PUFEM for the numerical solution of high-frequency elastic wave scattering on unstructured triangular mesh grids, Int. J. Numer. Methods Eng. 84 (2010) 330–350.

[47] O. Laghrouche, M.S. Mohamed, Locally enriched finite elements for the helmholtz equation in two dimensions, Comput. Struct. 88 (2010) 1469–1473.

[48] S. Li, W.K. Liu, Reproducing kernel hierarchical partition of unity, Part II—Applications, Int. J. Numer. Methods Eng. 45 (1999) 289–317.

[49] M.A. Schweitzer, Stable enrichment and local preconditioning in the particle-partition of unity method, Numer. Math. 118 (2011) 137–170.

[50] K. Pitchai, S.L. Birla, J. Subbiah, D. Jones, H. Thippareddi, Coupled electromagnetic and heat transfer model for microwave heating in domestic ovens, J. Food Eng. 112 (2012) 100–111.

[51] M. Rasoulpoor, M. Mirzaie, S.M. Mirimani, Electromagnetic and thermal analysis of underground power solid-conductor cables under harmonic and unbalancing currents based on FEM. Int. J. Numer. Model. Electron. Netw. Devices Fields 31 (2018) e2278.

[52] S.Y. Bae, G. Kang, J.K. Seo, Thermal and electromagnetic characteristics for cross-sectional design optimization of the integrated production umbilical, in: IEEE Energy Conversion Congress and Exposition, Montreal, Canada, 2015. https://doi.org/10.1109/ECCE.2015.7310173.

[53] S. Conti, E. Dilettoso, S.A. Rizzo, Electromagnetic and thermal analysis of high voltage three-phase underground cables using finite element method, in: IEEE International Conference on Environment and Electrical Engineering and IEEE Industrial and Commercial Power Systems Europe, Palermo, Italy, 2018. https://doi.org/10.1109/EEEIC.2018.8525354.